Ss. Cyril and Methodius University in Skopje
FACULTY OF COMPUTER
SCIENCE AND ENGINEERING

2019

Proceedings of the 16th International Conference for Informatics and Information Technology

Held at Hotel Bistra, Mavrovo, N. Macedonia
10-12 May, 2019

Editors:
Milos Jovanovik
Panche Ribarski

ISBN 978-608-4699-09-5

Conference for Informatics and Information Technology 2019
Website: http://ciit.finki.ukim.mk
Email: ciit@finki.ukim.mk

Publisher:
Faculty of Computer Science and Engineering, Skopje, N. Macedonia,
Ss. Cyril and Methodius University in Skopje, N. Macedonia
Address: Rugjer Boshkovikj 16, P.O. Box 393, 1000 Skopje, N. Macedonia
Website: http://www.finki.ukim.mk/
Email: contact@finki.ukim.mk

Proceedings Editors:

Milos Jovanovik
Panche Ribarski

Technical editing: Milos Jovanovik and Panche Ribarski

Cover page: Vangel Ajanovski

Total print run:150
Printed in Skopje, N. Macedonia, 2019

ISBN: **978-608-4699-09-5**

CIP - каталогизација на публикација
Народна и универзитетска библиотека „Св.Климент Охридски“, Скопје

004.7:621.39(062)
004(062)

PROCEEDINGS of the 16th Conference for Informatics and Information Technology (16; 2019; Mavrovo) Proceedings of the 16th Conference for Informatics and Information Technology: CIIT 2019, May, 10-12 Mavrovo, N. Macedonia / editors Milos Jovanovik and Panche Ribarski. - Skopje : Faculty of Computer Science and Engineering, 2019. - 164 стр. : граф. прикази ; 30 см

Библиографија кон трудовите

ISBN 978-608-4699-09-5
1. Jovanovik, Milos [уредник] 2. Ribarski, Panche [уредник]

Preface

This volume contains the papers presented at the 16th International Conference for Informatics and Information Technology (CIIT 2019) held on May 10-12, 2019 in Mavrovo, North Macedonia. The conference was organized by the Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, Republic of North Macedonia.

In the sixteenth edition, the aim of the CIIT conference remained to provide an opportunity for young researchers to present their work to a wider research community, but also facilitate multidisciplinary and regional collaboration. Despite the participation of scientists from the country, a substantial number of participants from abroad attended the conference as well. Building on the success of the past fifteen conferences, this year the conference attracted a large number of submissions resulting in presentations of 15 full papers, 9 short papers and 11 student papers, which were presented in five regular sessions and two student sessions. Traditionally, the conference included student sessions dedicated to the work of the best undergraduate students, where they presented their ongoing work, or demonstrated practical implementations. Three best student papers were awarded. The format of the conference allowed the participants to attend most of the talks that covered a diverse spectrum of research areas.

We had the pleasure to host three invited speakers. Blagoj Delipetrev, PhD, a scientific officer at the B6 Digital Economy Unit, Joint Research Center at the European Commission gave a talk titled "Artificial Intelligence: A European Perspective". We had three invited speakers covering different areas of the conference. Louiza Papachristodoulou, PhD, a Senior Researcher at NavInfo Europe B.V. performing research in cybersecurity for the automotive industry, gave a talk on "Side-Channel Attacks on Cryptographic Implementations and Efficient Countermeasures". Gjorgji Strezoski, a PhD candidate in the Intelligent Sensory Information Systems group at the University of Amsterdam, Netherlands, gave a talk on "Featurewise Transformations in Multi-Task Learning for Vision".

Part of the conference success is owed to the support received from our partners and sponsors: Ss. Cyril and Methodius University in Skopje, ICT-ACT, Loka, Medical IT Revolution, Sorsix, S&T, Bovin Winery and Pathfinder.

It is our belief that this year the CIIT conference successfully continued on its path to provide an opportunity for young researchers to present their scientific progress, as well as to bring researchers together. This paves the way for establishing a fruitful collaboration between disciplines, and confirms the conference's role as a testing ground for innovative ideas and for engaging the wider academic community.

September, 2019
Skopje

Milos Jovanovik
Panche Ribarski

Organization

Conference Chairs

Milos Jovanovik	Assistant Professor - Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Panche Ribarski	Assistant Professor - Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia

Organizing Committee

Biljana Tojtovska	Assistant Professor - Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Riste Stojanov	Assistant Professor - Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Petre Lameski	Assistant Professor - Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Bojana Koteska	Assistant Professor - Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Monika Simjanoska	Assistant Professor - Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Kostadin Mishev	Teaching Assistant - Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Aleksandar Stojmenski	Teaching Assistant - Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia

Program Committee

Ackovska Nevena	Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Ajanovski Vangel	Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Angelova Mihaela	INSERM, France

Antovski Ljupcho	Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Armenski Goce	Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Baicheva Tsonka	Institute of Mathematics and Informatics, Bulgarian Academy of Sciences, Bulgaria
Bakeva Verica	Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Balaz Antun	Institute of Physics, University of Belgrade, Serbia
Basnarkov Lasko	Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Bogdanova Galina	Institute of Mathematics and Informatics, Bulgarian Academy of Sciences, Bulgaria
Borissov Yuri	Institute of Mathematics and Informatics, Bulgarian Academy of Sciences, Bulgaria
Božinovski Adrijan	School of Computer Science and Information Technology, University American College Skopje, N. Macedonia
Chorbev Ivan	Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Davcev Danco	Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Delchev Konstantin	Institute of Mathematics and Informatics, Bulgarian Academy of Sciences, Bulgaria
Delibašić Boris	Faculty of Organizational Sciences, University of Belgrade, Serbia
Dimitrievska Ristovska Vesna	Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Dimitrova Vesna	Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Dimitrovski Ivica	Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Donchev Ivaylo	St. Cyril and St. Methodius University of Veliko Turnovo, Bulgaria
Eftimov Tome	Jožef Stefan Institute, Slovenia
Gievska Sonja	Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Gjoreski Hristijan	Faculty of Electrical Engineering and Information Technologies, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Gjorgjevikj Dejan	Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Gligoroski Danilo	Norwegian University of Science and Technology, Norway
Gramatikov Sasho	Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Gusev Marjan	Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Haller Stephan	Bern University of Applied Sciences, Switzerland

Ilievska Natasha	Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Jakimovski Boro	Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Janeska-Sarkanjac Smilka	Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Janev Valentina	Mihajlo Pupin Institute, Serbia
Jovanov Mile	Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Jovanovik Milos	Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Kalajdziski Slobodan	Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Kitanovski Ivan	Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Kon-Popovska Margita	Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Kostoska Magdalena	Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Koteska Bojana	Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Kulakov Andrea	Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Lameski Petre	Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Loshkovska Suzana	Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Madevska Bogdanova Ana	Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Madjarov Gjorgji	Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Marinkovic Bojan	Mathematical Institute of the Serbian Academy of Sciences and Arts, Serbia
Markovski Smile	Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Mihajloska Hristina	Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Mihova Marija	Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Mileva Aleksandra	Faculty of Computer Science, Goce Delchev University in Shtip, N. Macedonia
Mirceva Georgina	Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Mirchev Miroslav	Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Mishkovski Igor	Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia

Naumoski Andreja	Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Pachovski Veno	School of Computer Science and Information Technology, University American College Skopje, N. Macedonia
Papachristodoulou Louiza	Radboud University, Netherlands
Paprzycki Marcin	Systems Research Institute, Polish Academy of Sciences, Poland
Pepik Bojan	Max Planck Institute for Informatics, Germany
Popeska Zaneta	Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Popovska-Mitrovikj Aleksandra	Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Prckovska Vesna	QMENTA, Spain
Ribarski Panche	Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Ristanoski Goce	Data61, Commonwealth Scientific and Industrial Research Organisation, Australia
Ristov Sasko	University of Innsbruck, Austria
Samardjiska Simona	Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Sedigh-Sarvestani Sahra	Missouri University of Science and Technology, USA
Sherif Mohamed Ahmed	Data Science Group, Paderborn University, Germany
Shtrakov Slavcho	South West University, Bulgaria
Shurbevski Aleksandar	Kyoto University, Japan
Simjanoska Monika	Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Slavkovik Marija	University of Bergen, Norway
Spasov Dejan	Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Stoimenova Eugenia	Institute of Mathematics and Informatics, Bulgarian Academy of Sciences, Bulgaria
Stojanov Riste	Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Stojkoska Biljana	Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Stojkovikj Natasha	Faculty of Computer Science, Goce Delchev University in Shtip, N. Macedonia
Tojtovska Biljana	Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Trajanov Dimitar	Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Trajanovski Stojan	University of Amsterdam, Netherlands
Trajkovik Vladimir	Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Trivodaliev Kire	Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Trojacanec Katarina	Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia

Varbanov Zlatko	St. Cyril and St. Methodius University of Veliko Turnovo, Bulgaria
Zdraveski Vladimir	Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Zdravevski Eftim	Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Zdravkova Katerina	Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Zhelezova Stela	Institute of Mathematics and Informatics, Bulgarian Academy of Sciences, Bulgaria

Table of Contents

Full Papers

Short Papers

Student Papers

FULL PAPERS

Adopting Semantic-Driven Blockchain Technology to Support Newcomers in Music Industry

Nenad Petrovic
University of Nis,
Faculty of Electronic Engineering
Nis, Serbia
nenad.petrovic@elfak.ni.ac.rs

Abstract— **In recent years, music industry has dramatically changed, especially the way the music is delivered to the listeners. However, information technology has always been a double-edged sword for the music industry. Despite the fact that it made music creation much easier, from the other side, it has introduced many inefficiencies within the supply chain. In such conditions, there are many intermediaries between the music creators and consumers, making various aspects in music industry more complicated, such as revenue calculation, revenue share between many authors, reward delay, copyright and licensing issues. While established major artists collect most of the revenue from live performances, it is becoming more and more difficult for the newcomers to break through and earn from the effort they put in making music. In this paper, we analyze the modern music industry supply chain from the perspective of internet-based independent music artists and identify potential scenarios that could benefit from the adoption of blockchain technology. As an outcome, we propose a semantic-driven framework for automated Ethereum smart contract generation to support various scenarios with goal to make the existing platforms more flexible.**

Keywords—blockchain, Ethereum, music industry, supply chain management

I. Introduction

Since the breakthrough of Bitcoin cryptocurrency in 2008, blockchain has been considered as one of the most influential emerging technologies of the last decade [1, 2, 3]. Back then, it enabled the transfer of financial assets without spending additional money on an intermediary worldwide.

Quickly, a large community has been built around blockchain technology enthusiasts (both professionals and hobbyists) which led to the development of not only new generation of cryptocurrencies, but also consideration of many novel use cases apart from its application in financial transactions. From logistics and government to healthcare, there have been many tries to adopt blockchain in order to increase safety and trust of the existing information systems and applications.

When it comes to music industry, the rise of World Wide Web in last few years of the previous century has had a huge impact. It has not only revolutionized the process of music distribution, but also the way it is created. Before the internet era, music was physically distributed (vinyl, tape, CD) and sold in retail music shops, which gave the record labels control over the supply chain [4, 5]. However, the popularization of internet has made physical music distribution almost irrelevant. The two main turning points were the launch of online MP3 stores and later, switching to on-demand streaming platforms [5]. On the other side, affordable personal computers, audio hardware (high performance, low-latency DSP chips, MIDI controllers and USB microphones), software tools (virtual instruments, digital audio workstations) and a plenty of downloadable audio-related assets (loops, sample libraries) have made music creation process more comfortable and rapid than ever [6]. Moreover, the rapid information exchange enabled by internet together with piracy and peer-to-peer file sharing has brought many issues and challenges, as well, especially when it comes to revenue coming from intellectual property (where also music belongs) [7, 8].

Despite the advance in technology, the music industry is in decline compared to pre-internet and early days of internet era. Since 1999, music industry has almost been in constant decline, reaching the lowest point in 2014 [8, 9]. According to [4, 5, 8, 10], the main issues in current music industry supply chain are: 1) lack of industry-wide standardization for ownership of music releases 2) restricted access to transactional information 3) inefficient royalty payments due to many intermediaries involved. Due to blockchain's shared and decentralized nature, giving the ability to register, confirm and transfer any kind of property (beside money) without intermediaries, it can be considered as a potential solution to the mentioned issues.

In this paper it is explored how music industry can benefit from the adoption of blockchain technologies within its supply chain, especially when it comes to breakthrough of newcomers and emerging artists. As an outcome, a framework based on Ethereum blockchain technology is proposed leveraging the semantic representation of contracts and negotiation parameters between different participants for automated generation of corresponding input forms and smart contract code in order to provide more flexible support for complex revenue share scenarios and interoperability.

II. Related Work

As blockchain technology is being used for more than a decade already, there have been many tries to adopt it within the music industry supply chain with different goals. In Table I, the most significant among the existing blockchain-based solutions are presented together with their main goals and features.

TABLE I. Overview of blockchain-based solutions targeting msuic industry

Solution	Goals and features
Open Music Initiative[1]	To define an open source protocol that will enable uniform identification of music creators and rights holders by defining interfaces for achieving interoperability.

[1] https://open-music.org

Solution	Goals and features
PeerTracks[2]	Enable earning money from music streaming and simplify licensing of musical works.
Musicoin[3]	Streaming platform promising fair compensation, transparent contracts and no intermediaries in order to support independent artists.
Resonate[4]	Music streaming platform where musicians are paid directly through smart contracts.
Ujo Music[5]	Compensating artists and all collaborators using cryptocurrency and smart contracts.
Dot Blockchain Media[6]	An upgrade of existing music data standards that enables to maintain history of changes to a song's ownership using blockchain ledger.

In this paper, the main idea is to provide means for software enhancements that would increase the overall flexibility and usability of the existing solutions and technologies, rather than developing entirely new blockchain platform from scratch.

III. Background

In this chapter, the basics of blockchain, underlying mechanisms and related technologies relevant to the scope of this paper are presented.

A. Blockchain

Blockchain is a data structure (also called *ledger*) that consists of append-only sequence of elements (blocks) that store information about the executed transactions [2, 3]. The same term is often used for a distributed system that stores copies of the previously mentioned data structure within the peer-to-peer network of nodes. Each user (also called *node*) has its own alphanumeric address, which enables user anonymity ensuring transaction record transparency at the same time.

In blockchain, transaction itself represents transfer of value and ownership of digital tokens between sender and recipient that is recorded in the distributed ledger [1, 2, 3]. Tokens are used for the representation of both tangible and intangible assets – from cash and physical objects to copyrights, patents and intellectual property [2 ,3].

Moreover, each block also contains a cryptographic hash of the previous block and timestamp in order to ensure that no one can modify or delete them, once they are recorded in ledger. The more blocks are in a chain, the blockchain becomes more secure and reliable. Two types of blockchain networks are identified: public and private. Anyone can join public blockchain networks and each node maintains its own copy of the ledger. On the other side, in private networks, ledger is often permissioned so only authorized entities are able to act on a ledger.

When a new transaction occurs, it has to be validated and accepted by all the nodes within the network that act as miners and they are rewarded for the effort they put [1, 2, 3]. After the agreement, the ledger is in state of consensus. Several consensus protocols are accepted as standard in blockchain networks. Among them, widely used are Practical Byzantine Fault Tolerance (consensus based on majority) and Proof-of-Work, that is used in Bitcoin and Ethereum blockchain platform (based on computing effort instead of majority) [2, 3]. In order to hack the consensus, it would be necessary to create a whole new blockchain of modified records, which is an enormous and time-consuming task.

However, there are also some performance limitations that have to be considered when it comes to usage of blockchain. It is not suitable for storing data at high volumes or velocity as the data could be too large to be copied by each node, while the time and processing effort required for validation and verification of a block are often too high [5].

In this paper, we focus on adoption of open-source, public blockchain platform named *Ethereum*[7] [11] in music industry supply chain. We decide to use it due to its wide adoption in both scientific and commercial projects, while comprehensive documentation and literature about it is available. Ethereum platform application is referred to as *DApp* (decentralized application) that consists of both the frontend and backend code. The frontend code is often written in general-purpose programming languages, such as JavaScript, while the backend code (referred to as a *smart contract*) is often written in a higher-level domain-specific language.

B. Smart contract

Smart contract is a protocol intended to digitally facilitate, verify, or enforce the negotiation or performance of a contract [2]. In context of blockchain technology, smart contract is a software code that defines and executes transactions on the target blockchain platform, while the performed transactions are trackable and irreversible [ref]. Its distinctive feature is the ability to enable the execution of credible transactions without involving third parties.

A smart contract consists of business logic definition and operations that affect the state of the blockchain ledger in order to modify ownership and value of assets (represented as digital tokens) [2, 11]. When it comes to implementation, it can be written in any programming language and secured using encryption and digital signing. In case of Ethereum, smart contracts are written using a high-level object-oriented language Solidity[8], developed by Ethereum Foundation. It is far more expressive and powerful than Bitcoin's script language originally used for smart contract definition.

Despite the fact that Solidity seems quite similar to JavaScript, it also includes additional features that are used to support the implementation of transaction mechanisms in distributed environment of Ethereum blockchain network. It uses the concept of class for the representation of smart contracts. Similar to other object-oriented languages, instances of Solidity smart contracts contain fields and methods. While fields represent the state of a contract, methods represent the contract-specific operations that are be invoked in order to perform the transaction. However, when uploaded to Ethereum network, smart contracts are translated to lower level bytecode executed by the Ethereum Virtual Machine (EVM), while each node contains the same copy of that contract. Once a smart contract enters the blockchain it cannot be removed.

[2] https://peertracks.com
[3] https://musicoin.org/
[4] https://resonate.is
[5] http://www.ujomusic.com
[6] http://dotblockchainmedia.com
[7] https://www.ethereum.org/
[8] https://solidity.readthedocs.io/en/v0.5.5/

In this paper, smart contracts are considered as means to define the transaction process between the parties involved in the music industry supply chain. The idea is to enable the automated generation of smart contracts using semantic-driven framework, based on the agreements between the involved parties. Similar approach of automated smart contract generation based on ontologies was presented in [12], illustrated by two examples – one from healthcare domain and another one about car rental.

IV. Framework Overview

In this chapter, an overview of blockchain adoption in music industry supply chain is presented from perspective of semantic-driven framework for automated, flexible smart contract generation based on party agreement. Moreover, it is analyzed how the presented mechanism can be adopted in music industry supply chain from perspective of an independent, internet-based artists.

A. Automated code generation based on semantic smart contract representation

The existing applications of third parties involved in music industry supply chain (such as music distribution platforms and audio sample stores) offer only fixed pricing /royalty share schemas that are often not customizable (or customization is quite limited) which might not be suitable for all the use cases. Due to nature of music industry and possible scenarios, the additional degree of freedom and freedom is often necessary.

For this reason, we introduce the mechanisms of automated generation of smart contract code that is used to define transactional operations between the involved parties. Our main motivation is to enable more flexible peer-to-peer transactions, based on mutual agreement without any third parties involved.

The smart contract generation is divided into several phases (illustrated in Fig. 1). First, the parties negotiate about the contract parameters. The topics of negotiation could be of various nature, such as pricing, percentage of royalty share, duration of contract and many others. Once the parties agree on the contract parameters, the collected information is stored within the semantic knowledge base.

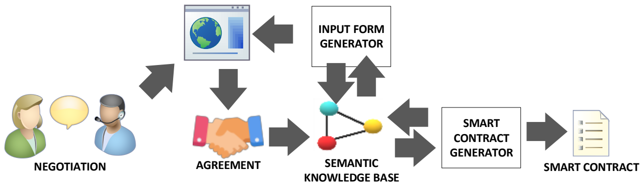


Fig. 1. Automated code generation based on semantic smart contract representation

After that, the knowledge about the party agreement is used to perform the automatic generation of smart contract code compatible with the target blockchain platform. During the code generation, smart contract template parameters are populated by the results of the SPARQL queries executed against the semantic triple store. If parameter value belongs to the specific business case, then additional code is appended to the template code to cover the particular case. Otherwise, the parameter value is just inserted within the template. This way, it is possible to have blockchain technology-independent description of the agreement which can be later used for generation of smart contracts supported by other platforms (not only for Ethereum). It is not only that we make a step towards interoperability between different blockchain platforms, but it also makes the smart contracts future-proof, which is of particular importance as blockchain technology is constantly evolving and there many emerging solutions. However, blockchain interoperability is still an open issue and there are many aspects to be considered in order to fully achieve it [13]. The smart code generation algorithm leveraging the knowledge stored within the semantic triple store is given as pseudocode in Listing I.

Listing I. Pseudocode of algorithm for smart contract code generation using the results of SPARQL queries

```
Input:  contract id, contract template, semantic knowledge base
Output:  smart contract for target blockchain platform
Steps:
For each contract parameter
        parameter.value:=query(parameter.name, contract id);
        if (parameter.value is specific case)
            then generate contract template code;
        else
            insert parameter.value into contract template;
end for each
end
```

Moreover, the semantic representation of contract types between different types of participants can be also leveraged to perform automatic generation of input forms that have to be filled in by the involved parties, using an algorithm given in Listing II. In most cases, the generated forms would be HTML forms with text input fields, checkboxes, drop-down lists and buttons. This way, the effort needed to generate new HTML is eliminated each time the support for new type of contract has to be added within the platform.

Listing II. Pseudocode of algorithm for input form generation leveraging the semantic representation of smart contracts

```
Input:  contract type, semantic knowledge base
Output:  HTML or other form
Steps:
Retrieve all parameters for specific contract type
For each contract parameter
        if(parameter is user-defined)
            generate text input;
        else
            if (parameter has one possible value)
                generate option elements for all possible values;
            else
                generate checkboxes for all possible values;
end for each
end
```

B. Semantic represenatation of smart contracts

Ontology represents the shared conceptualization of a particular domain. RDF[9] is often used for the representation of semantic data. It consists of classes, their properties and relationships expressed in forms of triplets (*subject, predicate, object*). SPARQL[10] is used for querying the RDF semantic triple stores. The knowledge within the semantic triple store used for smart representation is organized according to the ontology schema whose excerpt is shown in Fig. 2.

[9] https://www.w3.org/RDF/

[10] https://www.w3.org/TR/rdf-sparql-query/

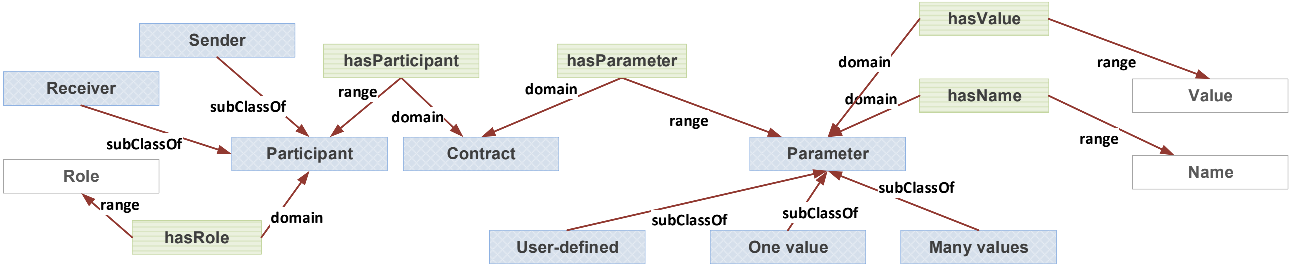


Fig. 2. An excerpt from ontology for semantic representation of smart contracts

Each contract consists of participants and parameters. Participants correspond to the parties involved in music industry supply chain, while contract agreements are expressed as parameters, such as royalty share percentage, type of contract, duration of validity and price. In smart contracts, each participant can act as either sender or receiver.

C. Smart contract adoption within music industry supply chain

Relevant roles and parties in music industry supply chain from perspective of internet-based independent artists are identified, based on [4] and [8] with some extensions (illustrated in Fig. 3): 1) Producer/songwriter: Creates and copyrights the composition, lyrics and/or arrangement. 2) Performer: Contributes to the recording by providing vocals or playing instruments. In most cases, when it comes to independent artists, they also perform the composition themselves. 3) Sample/loops distribution: Companies that sell sound loops and samples that could be single-shot sound effects, fragments or whole backing tracks. For newcomers, the usage of samples allows them to quickly achieve results. However, the massive usage of same sample packs by various artists and fair royalty share in this case are quite challenging. 4) Publisher: Entity whose role is to promote composition and its usage via many distribution channels (streaming platforms, CD stores). However, standard practice is that songwriters have to transfer ownership of their copyright to a publisher for exchange. Publisher is optional is supply chain, as some songwriters are acting themselves as publishers. 5) Rights society: A party whose task is pursuing royalty payments for every performance and usage of composition, anywhere in the world. 6) Distributor/Aggregator: An actor in music industry supply chain responsible for logistics - either for physical products (CD, vinyl) or digital distribution (via streaming services). 7) Consumption: A process of utilization of a published music work in many ways – streaming, live performance, using as background track, remixing, buying CD or vinyl.

The main idea is to leverage smart contracts in order to record the transactions between the involved making them transparent and immutable. This way, it is possible to share revenue according to established agreements, while publishing process can also be improved when it comes to licensing and usage of other authors' music samples. In what follows, the process of music publishing leveraging smart contracts is going to be described (illustrated in Fig. 4 in BPMN notation).

The first step is to establish agreement (contract) with all the involved music artists (producers/writers and performers). Moreover, it is necessary to make the contract for usage of music loops with sample distributor if there are any. After that, the complete music recording is submitted to publisher for review. During the review phase, it is needed to check whether the contracts with all the involved artists exist and are valid. For that purpose, the submitted music file has to be analyzed in order to detect matching with samples from loop libraries. If there is a match, the next step is to check whether the artist has made a contract for usage of the detected loops with sample distributor by checking the blockchain ledger of transactions. In case that artist is using unlicensed content, the submitted component is rejected. Otherwise, the process goes into the final phase where the contracts with other involved artists are checked. Finally, if all the necessary contracts exist, the submission is accepted and forwarded to distribution platforms in order to be consumed.

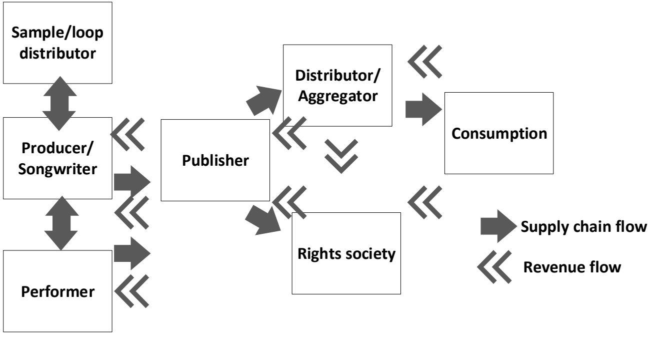


Fig. 3. Music industry supply chain and revenue flow from perspective of internet-based artists

V. Example Scenario

In this section, an example scenario illustrating the usage of the presented framework for ontology-based smart contract generator is given.

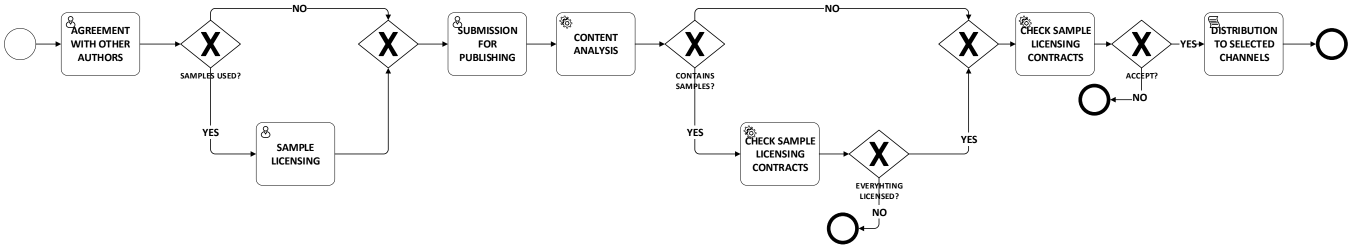

AGREEMENT WITH OTHER AUTHORS
SAMPLES USED?
NO
YES
SAMPLE LICENSING
SUBMISSION FOR PUBLISHING
CONTENT ANALYSIS
CONTAINS SAMPLES?
NO
YES
CHECK SAMPLE LICENSING CONTRACTS
EVERYHTING LICENSED?
NO
YES
CHECK SAMPLE LICENSING CONTRACTS
ACCEPT?
YES
NO
DISTRIBUTION TO SELECTED CHANNELS

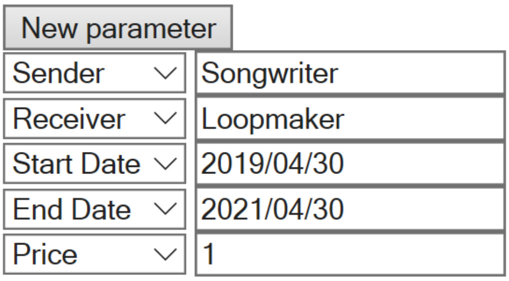
Fig. 4. Music publishing process based on smart contracts

LISTING III. RDF REPRESENTATION OF SMART CONTRACT IN EXAMPLE SCENARIO

```
<rdf:RDF
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:j.0="http://www.example.com/smcexample/"
    xmlns:j.1="http://www.example.com/tsc/resources/SmartMusicContract/" >
  <rdf:Description rdf:about="http://www.example.com/smcexample/Price">
    <j.1:hasValue rdf:datatype="http://www.w3.org/2001/XMLSchema#integer">1</j.1:hasValue>
    <rdf:type rdf:resource="http://www.example.com/smcexample/Parameter"/>
  </rdf:Description>
  <rdf:Description rdf:about="http://www.example.com/smcexample/Loopmaker">
    <j.1:hasRole>loopmaker</j.1:hasRole>
    <rdf:type rdf:resource="http://www.example.com/tsc/resources/SmartMusicContract/Receiver"/>
  </rdf:Description>
  <rdf:Description rdf:about="http://www.example.com/smcexample/EndDate">
    <j.1:hasValue>2021/04/30</j.1:hasValue>
    <rdf:type rdf:resource="http://www.example.com/smcexample/Parameter"/>
  </rdf:Description>
  <rdf:Description rdf:about="http://www.example.com/smcexample/Contract1">
    <j.1:hasParameter rdf:resource="http://www.example.com/smcexample/Price"/>
    <j.1:hasParameter rdf:resource="http://www.example.com/smcexample/EndDate"/>
    <j.1:hasParameter rdf:resource="http://www.example.com/smcexample/StartDate"/>
    <j.1:hasParticipant rdf:resource="http://www.example.com/smcexample/Songwriter"/>
    <j.1:hasParticipant rdf:resource="http://www.example.com/smcexample/Loopmaker"/>
    <rdf:type rdf:resource="http://www.example.com/tsc/resources/SmartMusicContract/Contract"/>
  </rdf:Description>
  <rdf:Description rdf:about="http://www.example.com/smcexample/Songwriter">
    <j.1:hasRole>songwriter</j.1:hasRole>
    <rdf:type rdf:resource="http://www.example.com/tsc/resources/SmartMusicContract/Sender"/>
  </rdf:Description>
  <rdf:Description rdf:about="http://www.example.com/smcexample/StartDate">
    <j.1:hasValue>2019/04/30</j.1:hasValue>
    <rdf:type rdf:resource="http://www.example.com/smcexample/Parameter"/>
  </rdf:Description>
</rdf:RDF>
```

Let us assume that an independent songwriter wants to use loops from the package produced by another artists (referred to as loopmaker). They negotiate about the price, license duration and distribution rights. In the end, they agree on the following contract conditions: the buyer can leverage the samples as much as he wants within the period of two years, while each commercial release containing the samples from that library will be charged 1 currency unit. After that period, the usage of samples is not possible.

The form used to parametrize contract is generated leveraging the previously presented ontology. The form used in this example is shown in Fig. 5. Users can add more parameters in order to materialize the agreement established once the negotiation ends by clicking the “New parameter” button.



Sender	Songwriter
Receiver	Loopmaker
Start Date	2019/04/30
End Date	2021/04/30
Price	1

Fig. 5. Contract form generated relying on semantic representation

The RDF representation of the smart contract inserted into the semantic triple store is given in Listing III. The generated smart contract is shown in Listing IV. The names of parameters whose values are obtained from the semantic representation are written in italic.

LISTING IV. SOLIDITY CODE OF THE GENERATED SMART CONTRACT IN EXAMPLE SCENARIO

```
pragma solidity ^0.4.21;
contract SampleLibrary{
      event Sent(address from, address to, uint amount);
      function send() public {
            if(now>StartDate && now<EndDate){
                  if (balances[Songwriter] < Price) return;
                  balances[Songwriter] -= Price;
                  balances[Loopmaker] += Price;
                  emit Sent(Songwriter, Loopmaker, Price);
            }
      }
}
```

VI. RESULTS AND EVALUATION

In this section, the experiments related to estimation of processing time necessary for code generation mechanisms are presented, considering both the smart code generation and input forms. The code generation was performed on a laptop equipped with Intel i7 7700-HQ quad core CPU running at 2.80GHz and 16GB of DDR4 RAM, while the RDF triple store was deployed on a Cloud server. The code generator is written in Java programming language and relies on Apache

Jena [11] for operations related to ontologies (RDF and SPARQL API).

In Table II, the obtained results using the previously described configuration and environment are presented. The first column shows the number of smart contract parameters involved. The second column displays the time needed for execution of SPARQL required for smart contract code generation process. The third column shows the time spent for smart code generation. The fourth column gives us information about the time spent for execution of SPARQL queries in case of input form generation, while the last column shows the time needed for generation of input forms. All the presented values are given in seconds and represent an average of 10 measurements.

TABLE II. CODE GENERATION TIME OVERVIEW

Number of parameters	***SPARQL queries [s]***	***Smart contract generation[s]***	***SPARQL queries [s]***	***Input form generation [s]***
3	1.94	1.23	2.17	0.81
5	2.21	1.51	2.68	0.94
7	3.12	1.75	4.17	1.12
10	4.19	2.31	5.12	1.18

According to the results, all the achieved processing times are order of magnitude of a second for up to 10 parameters involved. However, it can be noticed that SPARQL query execution lasts more than code generation procedures themselves. This can be explained by the fact that code generation was performed locally, while the queries were executed against the triple store deployed in Cloud. Moreover, it is obvious that input form generation needs less time, as mechanism for generation of HTML forms is simpler than the mechanism that involves treatment of specific cases for smart contract generation. From the other side, time spent for SPARQL queries is greater in case of HTML input form generator as the number of executed queries is greater due to fact that all the possible parameters need to be obtained, while for smart contract generation only selected values are necessary. Finally, the overall automated code generation procedure is faster, compared to manual code generation as its duration is estimated in minutes according to empirical results.

VII. CONCLUSION AND FUTURE WORK

In this paper, a framework leveraging the semantic representation of smart contracts in order to enable automated code generation that can make the existing blockchain solutions targeting music industry supply chain more flexible and extendable was presented. The fact that new types of smart contracts and corresponding input forms can be dynamically generated from semantic representation, gives the ability to extend the platform without making intervention to code directly but rather insert corresponding triplets for that purpose. This way, the additional time needed for deployment of new web pages could also be saved.

Despite the fact that adoption of blockchain within music industry supply chain seems quite promising, there are some obvious drawbacks, originating from the limitations of the applied blockchain technology itself. As number of transactions per second is quite low compared to other transactional systems, there are some use cases where blockchain's distributed ledger is a bottleneck. For example, the presented blockchain technology isn't suitable for tracking use of copyrighted material on pay per play basis [8].

Another consideration, when it comes to the semantic framework for automated smart contract generation presented in this paper is how and where to store the semantic descriptions of smart contracts. One possible solution is to keep the semantic data off-chain. In order to provide trust and ensure security in this case, additional effort is required for the implementation of corresponding mechanisms. Another solution could be to embed the semantic data within the blockchain. However, the mentioned issue is not covered by this paper and is considered as future work, together with mechanisms for analysis of the generated contracts in order to avoid attacks and misuse. The current source code of the implementation is available on GitHub[12] and will be updated in future.

REFERENCES

[1] A. Ushmani, "Blockchain Insight", International Journal of Computer Science Trends and Technology (IJCST), vol. 7 issue 2, March-April 2019, pp. 1-3, 2019.

[2] N. Balani and R. Hathi, Enterprise Blockchain: A Definitive Handbook, 2017.

[3] A. Narayanan and J. Clark, "Bitcoin's academic pedigree", Communications of the ACM, 60(12), pp. 36–45, 2017.

[4] C. S itonio and A. Nucciarelli, "The Impact of Blockchain on the Music Industry", R&D Management Conference 2018 "R&Designing Innovation: Transformational Challenges for Organizations and Society", pp. 1-13, 2018.

[5] O. Gough, "Blockchain: a new opportunity for record labels", International Journal of Music Business Research, vol. 7, no. 1, pp. 26-44, 2018.

[6] A. Lerch, "The Relation Between Music Technology and Music Industry", Springer Handbook of Systematic Musicology, pp. 899-909, 2018.

[7] O. F. Bustinza, F. Vendrell ‐ Herrero, G. Parry and V. Myrthianos, "Music business models and piracy", Industrial Management & Data Systems, 113(1), pp. 4−22, 2013.

[8] S. Derek and T. Seppala, "Digital Music Industry – background Synthesis", ETLA Working Papers, no. 48, pp. 1-22, 2017.

[9] Visualizing 40 Years of Music Industry Sales [Online], available on: https://www.visualcapitalist.com/music-industry-sales/. Last accessed: 27/03/2019.

[10] S. Zhao and D. O'Mahony, "BMCProtector: A Blockchain and Smart Contract Based Application for Music Copyright Protection", Proceedings of the 2018 International Conference on Blockchain Technology and Application (ICBTA 2018), pp. 1-5, 2018.

[11] A Next-Generation Smart Contract and Decentralized Application Platform [Online], available on: https://github.com/ethereum/wiki/wiki/White-Paper. Last accessed: 24/03/2019.

[12] O. Choudhury, N. Rudolph, I. Sylla, N. Fairoza, A. Das, "Auto-Generation of Smart Contracts from Domain-Specific Ontologies and Semantic Rules", 2018 IEEE International Conference on Blockchain, pp. 963-970, 2018.

[13] M. Borkowski et al., "Cross-Blockchain Technologies: Review, State of the Art, and Outlook" [Whitepaper], pp. 1-5, 2019. Available on: https://dsg.tuwien.ac.at/staff/mborkowski/pub/tast/tast-white-paper-4.pdf. Last accessed: 25/03/2019.

[11] https://jena.apache.org/

[12] https://github.com/penenadpi/SmartMusicContract

Advantages and disadvantages of monolithic vs. microservice architecture

Dajana Stojchevska
Faculty of Computer Science and Engineering
Ss. Cyril and Methodius University
Skopje, R. Macedonia
dajana.sk@hotmail.com

Gjorgji Madjarov
Faculty of Computer Science and Engineering
Ss. Cyril and Methodius University
Skopje, R. Macedonia
gjorgji.madjarov@finki.ukim.mk

Abstract—**The goal is to make an analysis of the advantages and disadvantages of the monolithic and microservice architecture through a simple use case scenario – book recommendation service. The use case will be represented first as a monolith and then as a microservice application in order to represent the key points that are identifying both solutions.**

Keywords—software architecture, monolith, microservices, domain-driven design

I. INTRODUCTION

Setting the software architecture has a key role in the software development process. The main purpose of the software architecture is to define the problems that might be encountered later in the implementation phase. If it is not well planned, even changing a small piece of code can cause other changes in other parts of the codebase. If there is a good architecture, then the final software product would be of a good quality. Although there is not a strict definition for a good architecture, there are several characteristics that if present in the software, are indicators of its good quality, e.g. easy to use, flexible, easily extensible for new features, scalable, works fast with high performances, reliable, easy to maintain, easy to refactor etc. Having an isolated, independent components in the code structure is a step forward into achieving these qualities.

The domain modelling is in the core of designing well-encapsulated components, which means that these components would not be affected from any implementation changes in other components. Having an isolated component guarantees its reusability as well, so in case where there are some general-purpose services (e.g. authentication service, payment service and similar) they should not contain domain logic and entities.

The software architecture of one application is mostly defined according to the business needs. The most popular architectures used nowadays are the monolithic and microservice architecture. Each has its own impact onto the productivity of the software developers and the quality of the product that is being delivered.

II. GENERAL COMPARISON

A. Monolithic architecture

Monolithic architecture is the traditional unified model that was used for designing software programs. Monolith means that everything is combined as one single giant piece. A lot of functionalities are living into a single service which is being tested, deployed and scaled, as a single unit. All its components within that service are interconnected and directly dependent of each other. That being said, monolith's components are low-cohesive.

Monoliths are very agile at the very beginning. The business could get a usable product way faster in comparison with microservices. It is way easier to set up an environment and right away start developing the monolith. It is also handy for setting up the cloud environment or configuring a server in order to deploy, because the codebase is kept in one place. Software developers' productivity in the beginning is on a high level. However, as the time passes by, the monolith is being built more and more, until it reaches a point where the developers' productivity significantly decreases as a result of the code complexity. The application's fast launching procedure comes with a technical debt because on the long run the developers would spend more time to make corrections on the existing code instead of implementing new features. Because of the huge codebase it is harder for a new team member to be involved fast into the team. Also, it is hardly possible to find a team member that is familiar with the whole codebase because mostly the team members know only some parts of it. All of this implies that it is very hard to split the developers into separate, independent teams. Each change into the code needs to be carefully reviewed and confirmed that it does not have any side effects, which indeed slows down the development.

The fact that monoliths persist as single service units brings them several advantages. For example, end-to-end testing can be executed easier because only one service is being tested. That eliminates the possible external errors that might arise, as they do when testing a distributed system (e.g. network issues). The communication between the classes takes place within a single process, which means it is fast and same as with testing, resistant to external impacts. Also, a single unit is easier to be deployed because there is only one directory to be handled. The state and its changes persist in a shared data model which is used by all parts of the application, so a monolith would not have any problems to keep a consistent state of the data.

B. Microservice architecture

The microservice architecture is the new trend in the latest years, used by many big companies offering global services to an enormous number of users, handling enormous amount of network traffic at a time. Microservices are the single units that are consisting the microservice architecture. Each microservice can be considered as a separate software product. A microservice has its own life cycle, which implies that software developers can build, test and deliver it independently of the other microservices. Each has its own specific task or purpose, accepts requests regarding that task, makes the needed processing and returns a response in a predefined format. It abstracts the implementation details and exposes only an interface that can be consumed in a consistent manner. Whenever making a change in the implementation of one microservice, only that part needs to be deployed and retested, not the whole application as in the monolith's case.

However, the investment in a microservice architecture is way bigger compared to a monolith. Setting up the environment to start working with microservices requires several segments from the DevOps sphere, like for example: continuous delivery, rapid application deployment, rapid provisioning, central logging, monitoring etc. All these technical challenges must be solved in the first few months of the software development, while the team is expected to deliver features. The above-mentioned technical solutions do not provide visual progress. Therefore, they have no business value, i.e. no function from the business aspect. It is similar with the deployment process. Each microservice requires a specially configured environment to run on. For this purpose, often the big companies like Amazon for example, have dedicated DevOps teams that take care of maintaining the servers, keep them up to date, keep them secure and other configurations. These services are intended to be used by other smaller companies, so the smaller businesses could focus on development and implementing their business logic.

Even though the investment in time and resources is bigger with microservice architecture, still microservices provide other advantages that can compensate for it, e.g. the development could be accelerated in different ways. Having in mind that the microservices have independent life cycle, independent teams can build them separately and work individually on their maintenance. Introducing parallelism in the work, of course speeds up the development. Besides, each team can choose a different technology stack according to the team competence and choose tools that best suit the business domain that is being solved. Microservices are also feasible for new team members because it is not important to know all microservices' implementation by heart, it is only important to care about their purpose, know how to make a use of their API and that is all. A new team member can only get involved into the implementation details of the microservice to which the team is dedicated. Microservices have a small scope, which is way better than having a huge codebase to cope with. When having a microservice architecture, a new functionality is most likely to be added as a new microservice, instead as a part of the existing codebase as is the case with monoliths. That saves a lot of precious time to the software developers. Instead of figuring out how to not break some other part of the code, they need to be careful only not to break the existing communication with the other microservices, or if doing so, to coordinate the change with the other teams. On the long run, even with the microservices the developers' productivity decreases, but still compared with the monoliths it would decrease much less.

Having a functionality as a separate service improves the testing in isolation. However, executing end-to-end testing might complicate the process if errors from external factors arise, e.g. network issues, hardware issues, environmental issues because services might be running on totally different machines etc., so there is a bigger chance that the communication between microservices will fail because of external not-controllable factors. All these make the debugging process more difficult as well. Not to mention that microservices might be in a different development phase, so when testing one, its dependency service might not even exist yet. However, a solution for these kinds of issues would be to prepare an API specification (e.g. using Swagger) and prepare some mock data in the predefined expected format. Anytime a dependent microservice data is needed, some mock data would be returned as a response instead, so the required microservice could be tested by assumption that this data is returned from its dependency.

Running as several individual applications brings several disadvantages to the microservices, for example the in-process communication present in the monolith's case does not exist here. The communication among the microservices is going to be slower and prone to external errors because they communicate through the network. That is acceptable having in mind that the microservices are running on different machines, most likely remote ones. Another challenge that the teams are facing with is how to cope with the distributed transactions. Each microservice would have its own data model. If executing some action means executing transactions across multiple microservices, a fault of any type might appear even in one of them, the required changes might not be applied everywhere or be applied partially, which will lead to an inconsistent state of the system. This can be solved with some event-oriented patterns such as event sourcing and CQRS (command query responsibility segregation) which help the teams to keep the data consistency in a distributed environment. Using these, the changes of the state which need to support the distributed transactions could be propagated as events (as with event sourcing) or as commands (as with CQRS). Each microservice participating in a transaction can afterwards subscribe to the event of interest.

III. Use case scenario – "My book"

In the upcoming sections, a simple use case scenario named "My book" will be used in order to analyze both architectures. "My book" represents a web application for recommending books. The user sees a list of books out of which only one can be chosen. By choosing one the user sees more details about it: title, author, genre/category and rate, as well as another list of similar (recommended) books. The user can give a rate to the chosen book. A similar book is defined as a book which belongs to the same category as the chosen one. The higher-rated books are on the top of the recommended books list. The use case diagram for this use case scenario is represented on Figure 1.

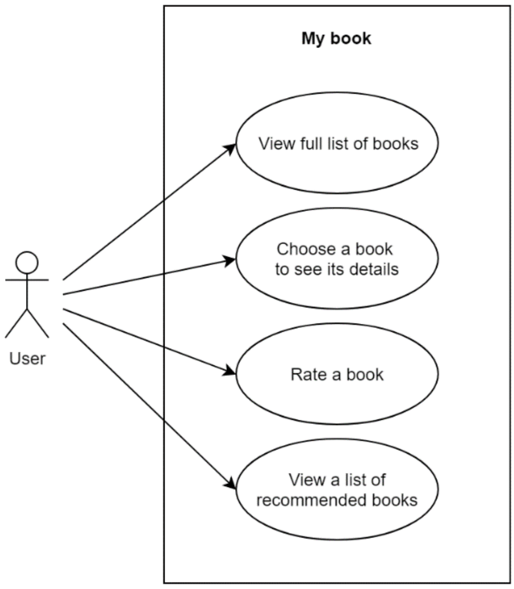

My book
View full list of books
Choose a book to see its details
Rate a book
View a list of recommended books
User

Figure 1 - "My book" use case diagram

The activity diagram for the use case scenario "My book" is represented on the following Figure 2.

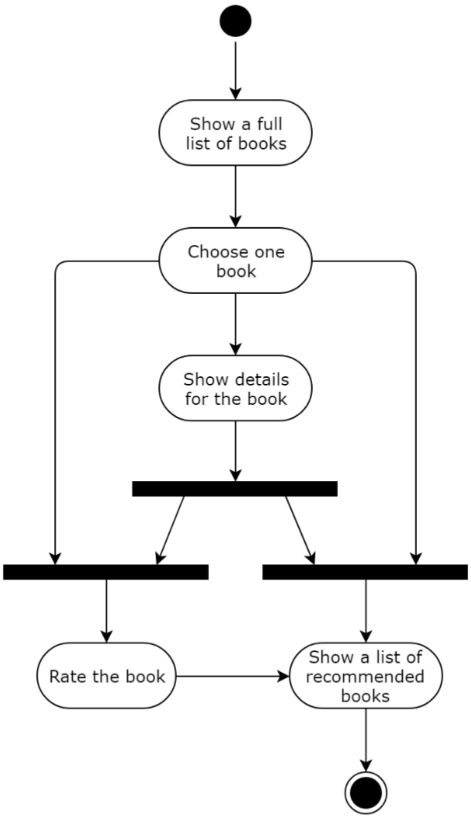

Show a full list of books
Choose one book
Show details for the book
Rate the book
Show a list of recommended books

Figure 2 - "My book" activity diagram

A. Analysis of "My book" as a monolith

The ideal architecture of one monolithic application would look as follows: several controllers, each communicates with its own service, each service uses its own data model and for each data model there is a dedicated repository. Everything looks good on a picture, but during development, the application slowly becomes a bunch of packages. For the application "My book" the code structure would then look as on Figure 3. The reason for this is something called "model-code gap" and is a result of a layered architecture. As it can be seen on the diagram there are layers and there is not a clear distinction which component is which.

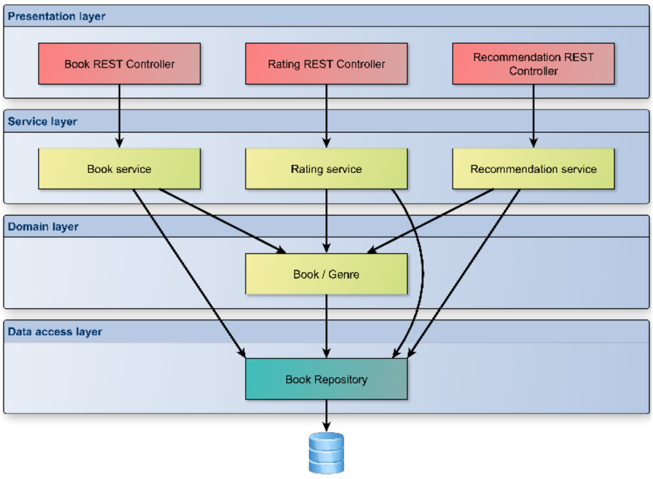


Figure 3 - "My book" with layered architecture

Regarding "My book", the presentation layer which would communicate with a client (mostly the user interface - UI) has exactly three elements: the controller that returns the full list of books with their details (*BookRestController*), the controller that accepts requests for giving a rate to some book (*RatingRestController*) and the controller that returns the recommended books list (*RecommendationRestController*). Each of these controllers has its own service which provides the data and processes the requests, but then the services communicate with a shared data model *Book*. This entity consists all information: a title, an author, a rate, a genre (category), so each service in order to process its request, it must take the data from this model's repository (*BookRepository)*. Everything is kept in one single model, shared by all components. Therefore, a component is a combination of several classes set in different layers. Although this eliminates any possible data inconsistency issues, this way of keeping the classes is not well-organized. It does not provide a clear evidence which controller uses which service, which service uses which model etc., coming to a situation where the components share common elements and they are coupled among each other. This is the main reason why the monolith's segregation into microservices could not be applied easily.

Often the decision for starting a new project with a monolithic architecture is based on the idea that the microservices' advantages would not be needed and used. The business is skeptical that there will be a lot of users for example, that this will decrease application's performances, or that some functionality would become used to a greater extent. At some point in future, when a bottleneck occurs somewhere (e.g. the server cannot handle that many requests at the same time), there is not an easy solution for the problem if the architecture is designed in layers. This kind of problem is quite serious and may damage the business. If the server fails to respond, the performances decrease significantly and the users (customers) can leave. The business must support larger number of users, must improve the existing quality and introduce new features in order to stay competitive on the market. One possible solution is scalability – clone the monolith's instance and set some load balancer to forward the requests to the adequate server instance. However, an optimal solution would be to take aside only that component of the application that is being the reason for the bottleneck (e.g. some newly added functionality), but a monolith could only be scaled horizontally, which means to clone the whole application as it is. Monoliths cannot provide any scaling which would be more optimal than that. As shown on Figure 4, the instance is cloned along with all application's functionalities, which results in wasting resources if the functionalities' usage is not equally distributed.

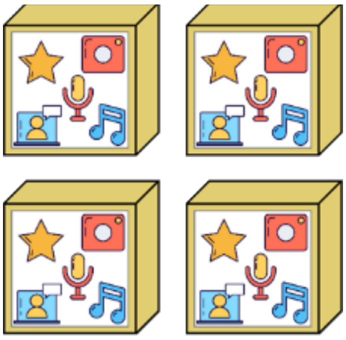


Figure 4 - Scaling a monolith

The basic problem that prevents monolithic applications from being scaled optimally is their architecture based on layers. It needs to be based on a services concept. A service represents one functionality, a job that needs to be done. Each service encapsulates one capability of the business, which the business is offering in a certain domain in order to fulfill its goals and responsibilities. The data and the services' functionalities are exposed through APIs (Application Programming Interfaces). So, if the application was based on such a concept only those server instances that are causing the problem would be cloned. Having in mind that "My book" is not functionally grouped, it can hardly be transformed into a microservice application and its desired functionalities to be scaled.

B. *Analysis of "My book" as a set of microservices*

Microservices allow the developers to focus on a specific problem from the domain. In "My book" there can be identified three concepts from the domain: reviewing books along with their details, rating a book and recommendation of books. Each of these could represent a separate component, which in future could be extracted and implemented as a microservice when there is a need for it. For example, if the number of users in future is raised so that the current running server instance cannot handle it and the users are looking for recommended books more often than giving a rate to some book, then this component for recommendation could be taken out as a microservice and cloned according to the needs. As an example, on Figure 5 can be seen how microservices provide more optimal way to spend the resources and scale only those components that appear to cause bottleneck. The central logging and monitoring will help identifying such components.

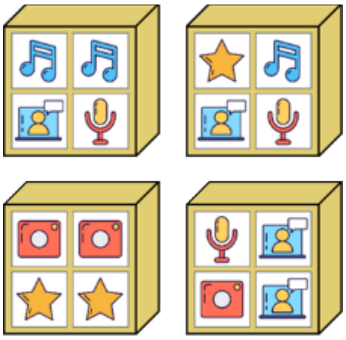


Figure 5 - Scaling microservices

The purposes and the tasks of the identified microservices are discussed below:

- The idea of the first microservice for books details would be to keep all the data available in the system for the books (title, author, genre) and to return these as a response. A note here must be given regarding the rating. The rate is not an information for a book itself because it comes from the users, therefore it should not be kept in here. Instead the service which is responsible for the rating will take care of it. In case there is a requirement for the rating to be shown as well along with the other details, then this microservice will communicate with the rating microservice and that is how it will show the rating, but never keep it on its own, because it is not its responsibility.

- The rating microservice would process all requests for giving a rate to some book and adequately update the average rate of that book. It must notify other microservices which are operating with average rates that there is an update in order the system to remain in a consistent state.

- The recommendation microservice would return a list of books which are similar to the book chosen by the user, sorted by their average rate. According to the definition for similar books in the use case description, the strategy for similar books would be implemented according to the category of the chosen book.

In order for "My book" monolith application to be able to break down into microservices, as the ones which are described above, its architecture should look as on Figure 6. Each component correspondent to the above-described potential microservices now represents its own completed set. Even from the diagram their independence could be noted.

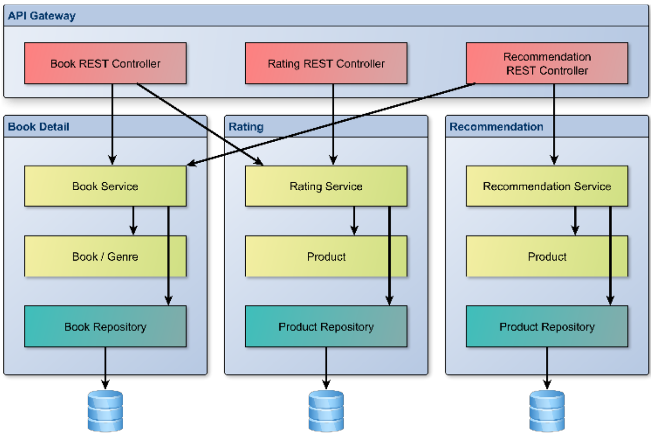


Figure 6 - "My book" with domain-driven architecture

The recommendation service does not communicate with the model *Book* anymore and with its repository, but it has its own model named *Product* and a repository for it, which means it has its own database. The same can be said for the rating service. This is still a layered architecture, but now the layers are implementation details of the components, rather than primary architectural building blocks. All three components are being used by another fourth component (API Gateway) which represents the controllers. This component API Gateway provides a unified interface and communication protocols with the services. It accepts all requests from the client application (UI) and takes care each request to be forwarded to the adequate service, in order to get the right response from it. Regarding the newly introduced entity *Product* and its relationship with the entity *Book*, the answer is given by an indispensable concept needed for working with microservices and that is domain-driven design (DDD).

DDD is a software development approach that has existed long before the appearance of microservices on the scene. However, its popularity has especially increased with the popularization of microservices because of its importance when working with them. DDD premises to put the project focus into the core of the domain and its logic. Ideally, it is preferred only one unified model to be created, but when working on a big project and several models come into play, combining the code based on different models makes the software prone to errors, not reliable and hardly understandable. Therefore, there is needed an explicit definition of the context in which each of the models is being applied. For these purposes there is a pattern called bounded context. The bounded context takes the central position into DDD because it serves for handling with many domain models. The models are being divided into several bounded contexts and their interrelationships are being highlighted. For each model there are bounders regarding its usage in some parts of the application, the code and the databases. Still there is a problem when taking a global view – one bounded context when seen on its own could be unclarified because the context of the other models might not be known yet. The people from other teams that are not aware of the context bounders could make changes that are blurring the bounders and are complicating the interconnections. Therefore, as a solution there is another DDD pattern called context map, consisting the bounded contexts and their connections.

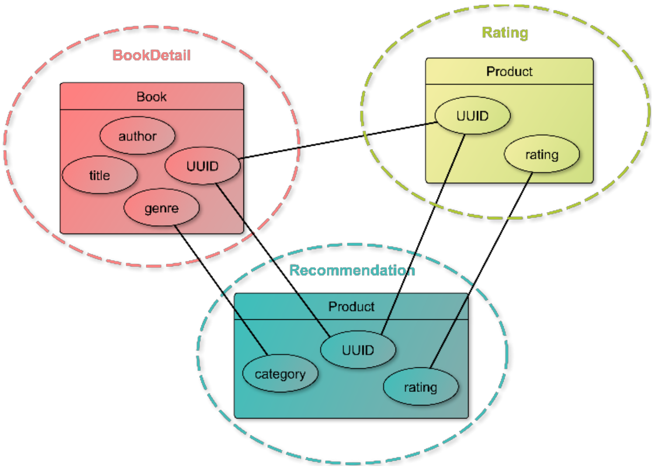


Figure 7 - Context map for "My book"

The context map for “My book” is shown on Figure 7. There is represented the relation between the entity *Book* which belongs to the bounded context of the concept for reviewing all books and their details. On the right side there is the entity *Product* in the bounded context of the concept for rating and at the bottom there is one more *Product* but this time within the bounded context for recommendation. Each of these entities serves for solving only the problem on which it is referring to, which is the *BookDetail*, the *Rating* and the *Recommendation*, respectively. Therefore, the attributes that *Book* has are author, title and genre, which is the only data that are needed so that the microservice for book details could do its job properly. For rating a book, it is enough to keep only the information about the average rate which means that only this attribute is needed as a field of this entity. The microservice for recommending books needs the exact category in which the chosen book belongs, in order for the similar books to be recommended as the business logic is defined. It is also using the average rate to sort the books.

The three entities are connected among themselves with a unique universal identifier (UUID). Instead of using standard primary keys (1, 2, 3…), there is used this type of identifier which actually represents a combination of alphanumeric characters. For one specific record, this value must be equal wherever this record is being used, that means in all tables and databases used for “My book”. Using this value all data about the specific record would be connected (joined), e.g. it will be known which book is adequate with which product.

Within an application with a structure like this one, the components are not communicating directly among themselves. The best way for them to communicate would be using asynchronous messages. For example, when the user gives a rate to some book, the component responsible for processing the request (the rating component in this case), sends a message to other interested components for this information. Afterwards those components (the component for recommending books in this case) need to read that message and update the adequate product in their own databases. If this communication does not happen, then the recommendations component would list the recommended books by some older average values, i.e. there is a data inconsistency in the system. If any kind of transaction is happening in some component, that component is responsible for emitting an adequate event for that action which arrives in all components subscribed for that event. The component which is emitting the event is called a publisher. Each of those components that are responsible for processing the published information, are updating their records. They are called subscribers. This pattern is very commonly used among microservices and is supported by message brokers.

IV. Conclusion

A good software architecture can help in providing a better estimate of the budget and the resources needed for the project. To define the architecture means to make decisions, which implies making compromises. For example, the microservices are more expensive to set up so the business might decide to go with a monolith instead, at least in the beginning. Even if the application is planned to be a monolith, the domain modelling is the first step that needs to be considered when setting the architecture of a new application. That guarantees that at some point later in time the desired component can be taken out of it and within several steps to be deployed as a separate application. The components in the design must be highly cohesive, which implies low coupling among them. This minimizes the possible “domino effect” which arises when making a change into the code.

Microservices can improve the development time, not only because of their simplicity and small scope, but because independent teams can work in parallel. Each microservice has its own lifecycle and can be scaled on its own according to the business needs. Therefore, whenever thinking to start with some application, its modularity and its encapsulated components reflecting a functionality must be in the focus of the design thinking. Otherwise, the application will end up as a “big ball of mud” having coupled components into its structure, hardly ever being able to use at least one of the microservices’ advantages.

References

[1] Hasselbring, W. and Steinacker, G., 2017, April. Microservice architectures for scalability, agility and reliability in e-commerce. In *2017 IEEE International Conference on Software Architecture Workshops (ICSAW)* (pp. 243-246). IEEE.

[2] Mazlami, G., Cito, J. and Leitner, P., 2017, June. Extraction of microservices from monolithic software architectures. In *2017 IEEE International Conference on Web Services (ICWS)* (pp. 524-531). IEEE.

[3] Hasselbring, W. and Steinacker, G., 2017, April. Microservice architectures for scalability, agility and reliability in e-commerce. In

2017 IEEE International Conference on Software Architecture Workshops (ICSAW) (pp. 243-246). IEEE.

[4] Balalaie, A., Heydarnoori, A. and Jamshidi, P., 2016. Microservices architecture enables devops: Migration to a cloud-native architecture. *IEEE Software, 33(3)*, pp.42-52.

[5] Gouigoux, J.P. and Tamzalit, D., 2017, April. From monolith to microservices: Lessons learned on an industrial migration to a web oriented architecture. In *2017 IEEE International Conference on Software Architecture Workshops (ICSAW)* (pp. 62-65). IEEE.

[6] Villamizar, M., Garcés, O., Castro, H., Verano, M., Salamanca, L., Casallas, R. and Gil, S., 2015, September. Evaluating the monolithic and the microservice architecture pattern to deploy web applications in the cloud. In *2015 10th Computing Colombian Conference (10CCC)* (pp. 583-590). IEEE.

[7] Vernon, V., 2013. *Implementing domain-driven design*. Addison-Wesley.

[8] Avgeriou, P. and Zdun, U., 2005. Architectural patterns revisited-a pattern language.

[9] Ebert, C., Gallardo, G., Hernantes, J. and Serrano, N., 2016. DevOps. *Ieee Software, 33*(3), pp.94-100.

[10] Dragoni, N., Lanese, I., Larsen, S.T., Mazzara, M., Mustafin, R. and Safina, L., 2017, June. Microservices: How to make your application scale. In *International Andrei Ershov Memorial Conference on Perspectives of System Informatics* (pp. 95-104). Springer, Cham.

Concepts of Vertical Mobility Vehicles and Autopilot Systems

Kiril Frenkov
Faculty of Computer Science and Engineering
Ss. Cyril and Methodius University
Skopje, Macedonia
kiril.frenkov@makedon.tech

Nevena Ackovska
Faculty of Computer Science and Engineering
Ss. Cyril and Methodius University
Skopje, Macedonia
nevena.ackovska@finki.ukim.mk

Abstract— **The objective of the presented work is to identify an early concept of personal vertical takeoff and landing (VTOL) urban air mobility (UAM) vehicle. Those vehicles most likely will operate autonomously. The paper aims to emphasize the complexity of the vehicle safety systems, the needs of efficient trajectory navigation through the urban area, while localizing the surrounding traffic and operating with low-level pollution. The current state-of-the-art technologies that could serve as a basis for building more efficient vertical mobility vehicles are also discussed.**

Keywords—Future urban air mobility (UAM), Autopilot and System Architecture, Artificial Intelligence (AI), electric vertical takeoff and landing (eVTOL) vehicles.

I. Introduction

With the existing modern infrastructure and mobility concept, many large cities are growing in size and are facing collapse of their transportation systems. Road space is a limited resource. Large number of personal motor vehicles most of the time are often parked on the roads, especially in high density inhabited city areas, and together with the public transportation they result with very slow-moving traffic and elevated local levels of pollution (in terms of noise and gas emissions). The complete adaptation of the existing city infrastructure toward the scale of the expected traffic is too expensive or simply not feasible. Therefore, some other concepts that ease the overcrowded infrastructure are already emerging. One of those concepts is the shared taxi vehicles concept, probably equipped with self-driving system, which could help to reduce the traffic and increase road safety. Other future possible ideas would be to spread the traffic toward the city sky and provide a third-dimension mobility or so-called urban air mobility (UAM).

The concept of the vertical mobility is taken in consideration in several studies. In study [1] the authors write about how the on-demand aviation has the potential to radically improve urban mobility. They supposed a small network of electric aircraft that take off and land vertically (eVTOL), to enable rapid, reliable transportation between suburbs and cities and within cities. The skyscraper and the parking garages tops, together with the existing helipads, as well as the unused land surrounding highway interchanges, are suggested as possible ground for basis of a distributed network of “vertiports” or single-aircraft “vertistops”. In order to be accepted by the market the VTOLs traffic control system and the autopilot must achieve top-level of safety similar to the airline aviation, which is crucial for public adoption. Study [2] presents the market potential observing three main use cases for passenger drone services – air taxis, airport shuttles and intercity flights. This study also identifies five basic electric aircraft architecture and trade-off which one is better suited to the different use cases. In study [3] VTOLs are projected to start providing commercial mobility services in 2025. They analyze the market size separated in three segments of vertical mobility services: aerial inspection, goods and passenger transport. The study includes different scenarios for intracity and even city-to-city trips, comparison of costs and travel time between using a road and aerial vehicles.

In order to acquire accreditation and permission to be used in public VTOLs will be manufactured, flown, and maintained to meet the stringent levels of control and supervision from the US Federal Aviation Administration (FAA) and European Aviation Safety Agency (EASA). Additionally, the on-demand aviation, at least until autonomous VTOL become reality, requires commercial pilots with higher level of training, experience, flight review, and medical certification.

In order to define some of the important aspects of the vertical mobility as a concept that is expected to become important in the following decade, we shall first discuss the elements of the current technology that could derive effective solutions for the vertical mobility. The maturity of the automated ground vehicles might prove very important in the vertical mobility solutions. Furthermore, the autopilot systems for the aerial vehicles also could provide some solutions for the concepts of the vertical mobility. Thus, this paper is organized as follows. The second section describes the idea and the ongoing work in the development of an automated ground vehicle concept. The maturation of this technology will be a big step toward the urban air mobility. The third section puts accent on the predevelopment of an intelligent autopilot system, safety concerns and certification. The fourth section presents a review of the concepts of electric vertical takeoff and landing (eVTOL) vehicles, describes the technology benefits and the industry convergence of technologies essential to develop an urban air vehicle. The last section presents the conclusions of this review.

II. Automated Ground Vehicles

Highly automated road vehicles sooner or later will become a reality and eventually improve the traffic safety, helping to save lives and prevent injuries. The ultimate challenges to build a self-driving system are mostly technical, but there are major legal, social, ethical and human issues to be resolved. Besides delivering great safety benefits, self-driving vehicles will also promote improved energy efficiency, decreased pollution and less congestion, while greatly facilitating mobility for the elderly and people with disabilities. Full societal benefits today are difficult to be projected and understood. Most likely, a driving experience enhanced with Internet of Things (IoT) will makes commuting safer, more productive, and enjoyable. Those

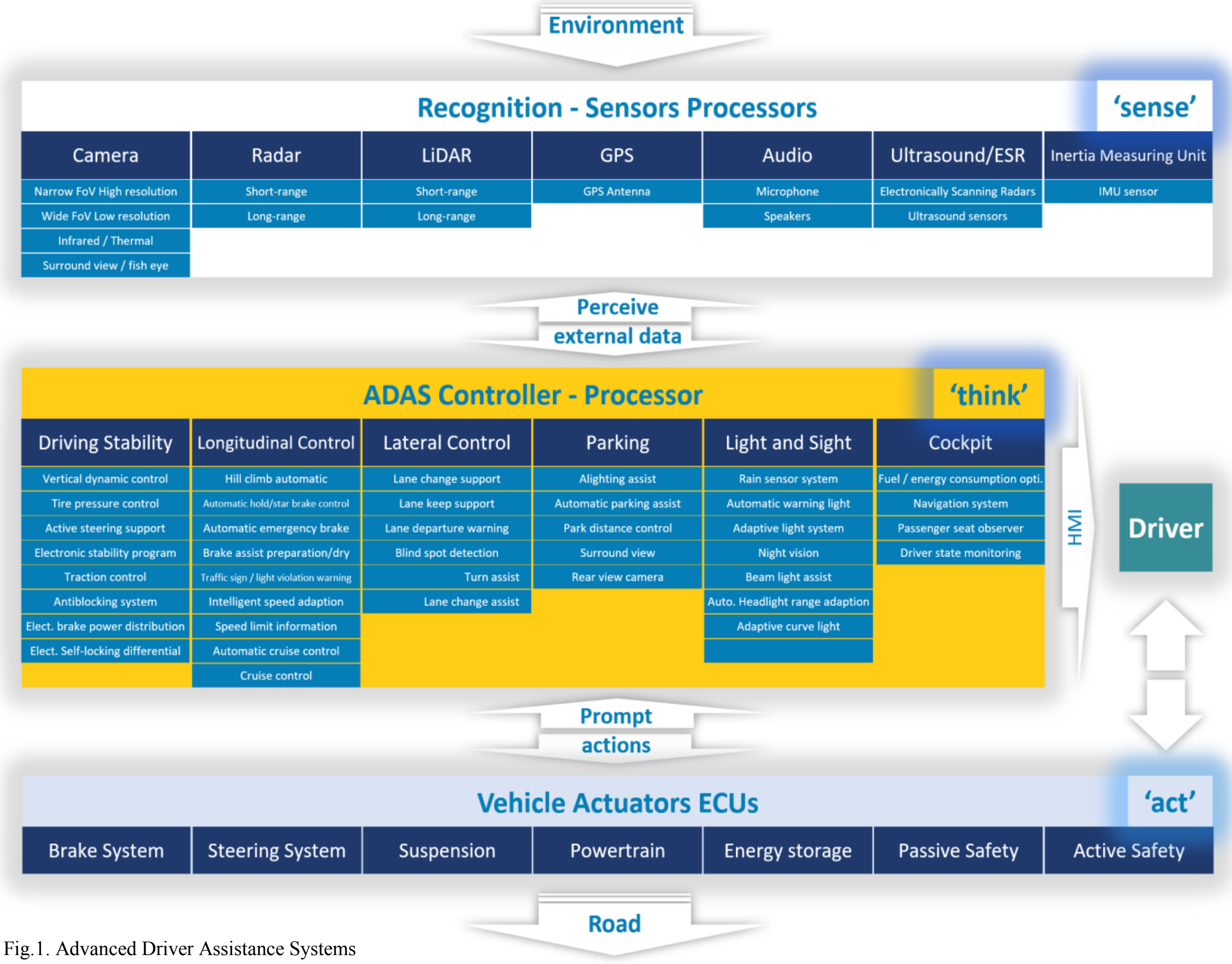


Fig.1. Advanced Driver Assistance Systems

undergoing radical transitions in the automotive industry present unique opportunities for the automotive and IT professionals to take fundamentally different approaches to vehicle function, design and construction.

Today's state-of-art advanced driver assistance systems (ADAS) have the potential to remove the human error in traffic accidents, which will help protect the driver and the passengers, as well as the pedestrians and bicyclists.

According to the Society of Automotive Engineers (SAE) the fully autonomous vehicles will integrate onto the roads by progressing through six levels of driver assistance technology advancement in the next few years [4]. Today we are on SAE level-3 maturity of ADAS technology, as shown on Fig.1, where we have embodied a lane change assistance system; adaptive cruise control; automatic parking and braking; blind spot detection; collision avoidance and many other functionalities to support the driver. Ultimately, SAE level-5 will label the vehicles without driver or a self-driving vehicle. Regarding the technical layer ADAS distinguished three major functional domains: environmental perception ('sense'), decision-making ('think') and control ('act'), as depicted on Fig.1.

Automated driving functionalities are the key know-how essentials for development of an intelligent vehicle. Smart driving capabilities enable the vehicles to constantly detect changes in their immediate surroundings, take proper driving logic decisions, and implement them in terms of steering, acceleration, and braking. An intelligent vehicle, with aid of reinforcement learning should be able to complete proper driving policy logic, causing proper strategy, tactics, executing the tactic decision with proper prediction of trajectory and acting through the controls.

Present-days Original Equipment Manufacturer (OEMs) automakers and Information Technology (IT) companies race to build the ultimate ADAS, but they differ in the approach and engineering philosophy. Both companies Waymo and Tesla [5] have the same goal to build a fully autonomous driving car. Tesla relies on computer vision systems that combine radar, camera and ultrasonic devices. Waymo, on the other hand, counts on a LiDAR (Light Detection and Ranging) sensors as most relevant in the system.

Tier-one suppliers also race to develop an intelligent add-on system. Fig. 2. presents an autonomous driving add-on equipment from Aptiv [6].

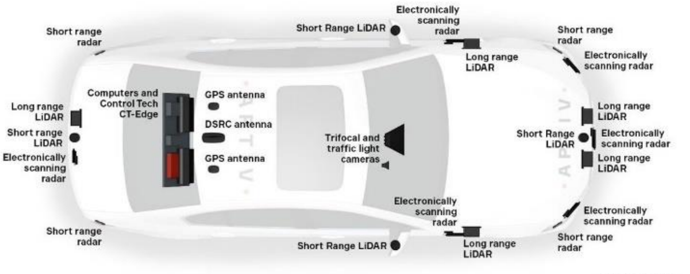


Fig.2. Add-on equipment from Aptiv [6]

Implementing all of those sensors and devices into modern car architecture, is not straight forward task, especially when it comes to packaging the typical Radom/ Lidom receiver-transceiver screens in to the vehicle facia or roof. Such sensors location barely intrudes the well-known brand styling traditional car design. Usually it is really difficult to convince and influence the car designer team to adopt the traditional car design and architecture towards the self-driving system sensor package build-up from the engineers. To develop a sensor package of an ADAS with correct functionality it is crucial to place all of the externally exposed sensors in such a way so they could be as much as possible self-cleaning from the environment pollution, to have wide range of Field of View (FoV), as well as to locate all the optical sensors on a dirt-free and impact-free vehicle exposed areas. Cemented salt crystal and accretion of the winter ice are particularly hard to clean with the car limited fluid resources, and both cases often reduce the sensor "transparence" to a level of system malfunction.

In the recent years semiconductor companies and other players have made important advances technology enhancements. ADAS application incorporates many technologies, and they take special care of processors, sensors, software algorithms, and mapping.

Electronic control units (ECUs) and microcontroller units (MCUs) are essential for most ADAS applications, including autonomous driving. For ADAS to advance, processors need better performance, which could be enabled by multicore architectures and higher frequencies, as well as lower power-consumption requirements.

Sensors have a limited measurement range and signal bandwidth, which makes them difficult to distinguish between "signal" and system "noise." It is especially difficult for the sensors to track moving objects during less-than-ideal environmental conditions, such as rain and fog. The manufacturers are attempting to optimize the system performance through better sensor fusion of coherent combination of data from multiple sensors. On the hardware side, intersensory communication is a major challenge because it requires high bandwidth and solutions for preventing network overloads. On the software side, the fusion of image and nonimage data is particularly challenging. Some OEMs and tier-one suppliers are working together with academia to address this challenge. The limited functionality of today's sensors, combined with their high cost, may be the greatest constraint to ADAS uptake. If camera-based solutions catch up to radar and lidar in functionality, they could eventually dominate the ADAS market because of their lower cost. "One box" solutions that combine lasers and cameras may also become popular because they are less expensive than radar or lidar alone. This is an important development, since experts believe that semiautonomous driving will not become a reality until the industry has a cost-effective lidar system that is fully integrated with other sensors. There is no single sensor modality that meets all the requirements of an autonomous vehicle.

Algorithms that are running on ECUs and MCUs, use the input from the sensors to synthesize the environment vehicle surrounding in real time. The algorithms then provide output to the driver or specify how the system should actively intervene in vehicle control.

In response to developments in sensor fusion, the industry is about to transition from embedded software running on a separate ADAS-specific ECU to software platforms running on centralized ECUs or MCUs. These software platforms have a higher level of abstraction to allow flexible integration of sensor-fusion algorithms. Industry players are now focusing on creating such algorithms, which allow for more accurate synthesis of sensor data and more efficient processing, because they will help prevent data overload or slowdowns. Another priority is creating algorithms that allow for safer car navigation and more accurately predict all possible human behavior—including potentially irrational responses—in various situations, such as when a collision between two cars appears imminent.

Detailed and accurate mapping systems can help prevent accidents when Global Positioning System (GPS) coverage fails, such as through tunnel driving. These systems also store geographical and infrastructure information, and communicate with onboard sensors to determine a car's exact location.

Connected cars and Internet of Things (IoT) technologies are focusing on vehicle message passing, in which the car communicates with: Other vehicles (V2V); Infrastructure points, like signs and traffic lights (V2I); Bicyclists and pedestrians, via smart-phone apps and wearables (V2P); The cloud (V2C) and finally vehicle-to-everything (V2x). The main challenge with V2x is what to do with the incoming data. Many proponents feel that the best approach is to compile and present it to the driver to interpret and act upon. Much industry work is required to ensure that V2x communication protocols are standardized. To get a serious systemic payoff from the technology, all vehicles must speak a common language. Disparate formats will both diffuse the safety benefits and confuse consumers. Finally, V2x by itself is dependent on robust network connectivity. Not all wireless communication protocols are created equally in ensuring secured, deterministic message passing. And not all can achieve global ubiquity, efficiency, and cost-effective economies of scale.

As one can observe, the above-mentioned features of the autonomous guided land vehicles also could apply for the automated vehicles in general. The low-level components and know-hows applied at the autonomous car system could be inherited in the flying urban area systems due to their similarities. They differ in precision and reliability, the need of redundancy and certification, and differences between the two applications mainly coming from the controlling complexity. Nevertheless, advance in autonomous cars technology with full automation is necessary but not sufficient to enable fully automated aircraft. Aerial specific development in the areas of risk awareness and management is still needed, as well as additional development of perception systems capable of cognation of local weather status and risks.

III. Intelligent Autopilot System

Automotive and IT company are competing to build fully autonomous driving system. The upcoming launch of this technology implemented into ground vehicle will urge an early development of an Intelligent Autopilot System (IAS) designed for urban air vehicles. The development of low-cost, safety-critical automotive systems will likely be highly applicable to safety-critical trajectory control and other system management functions.

Before we begin to describe the potentials to integrate ADAS to a VTOL we need to explore several significant differences between ground vehicles and small aerial passenger vehicles. Ground vehicles operate exclusively on a network of roads that are made to maintain a nominal level of risk and demands on the driver in almost all conditions, except weather hazards. For example, the performance of a ground vehicle is largely unaffected by road non-hazardous weather, such as strong winds and heavy rain, where the driver decreases the vehicle speed and pull over the vehicle to the side of the road for a period of time. Needless to say, such operational weather conditions largely impact the performance of urban VTOLs and could be referred as not safe or even become unacceptably risky. Upcoming IAS route planners must incorporate significant knowledge-based reasoning highly related to weather, to be able to assess potential hazards such as turbulence, hail and ice accumulation types and rates.

Airways are engineered to minimize some risks such as terrain hazards, radio navigation and communication blockages. Primary risks factors in vertical takeoff and landing maneuvers manly depend on the presence of other air traffic; birds singular and in flocks strikes; and weather phenomena.

Prior to route planning, an IAS must take responsibility for ensuring the fitness of the vehicle for the intended trip as well as ensuring proper loading and respecting gross weight limits, where the location of the aircraft's center-of-gravity must fall within specified limits to keep the stability and the control characteristic of the vehicle. Prior to operation the IAS will likely be required to perform self-inspection for fail-safe, built in checks, to assure safety for all basic vehicle properties.

Current operational autopilots fall under the domain of Control Theory. Classic and modern autopilots rely on controllers such as the Proportional Integral Derivative (PID) controller, and Finite-State automation [7]. Many recent research efforts focus on enhancing flight controllers by adding fault/failure tolerant capabilities.

Considering a typical fixed-wing airplane, the control tasks consist of managing a coupled, multi-input, multioutput system. Pitch (elevator), roll (aileron), and yaw (rudder) moment effectors are used in conjunction with a longitudinal-power to manage airspeed, vertical speed, and turn rate during most phases of flight and airspeed, vertical speed, lateral translation, pitch angle, and heading angle during landing. Cross coupling of an airplane to control inputs is typically much greater than a car. To steer a car, we simply rotate the steering wheel, while maneuvering an airplane quickly to a new heading involves simultaneous coordinated change of elevation, aileron, rudder, and throttle position.

The impact on incoming technology transfer, adopting the ground vehicles ADAS to personal VTOL, creates the opportunity to fundamentally change the role of the trained human pilot to an operator focused on mission and risk management, while relying on the automation to perform high-bandwidth, motion control and system management tasks.

Considering the intelligence of the autopilot systems, there are several models that have already been proposed. The next generation of airplane IAS's should integrate the following components: a flight simulator, an interface, a database, a flight manager program, and Artificial Neural Network (ANN) [8]. The IAS implementation method has three steps: pilot data collection, training, and autonomous control. The presented IAS is based on supervised learning, which means that the existing system is initiated by human pilot maneuvering the physical vehicle in real environment.

In [9] a reinforcement learning algorithm was successfully applied to a controller for autonomous helicopter inverted hovering. Using data collected from the helicopter in flight, a stochastic helicopter's nonlinear dynamics model by learning was built.

Most likely the next IAS generation will combine different AI tools such as deep reinforcement learning, deep Q-networks [10] or other new type of sequential decision-making algorithms where the goal is to learn how to act optimally in a given environment with unknown dynamics.

IV. Urban Air Vehicles

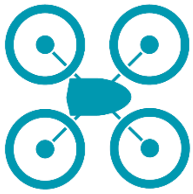
The idea of urban air vehicles has already emerged. There are several companies and many researchers that deal with the idea of using the urban air vehicles in the everyday living. In [1] UBER elaborated on the plans for an urban air transportation service using thousands of eVTOL aircrafts – eventually autonomously piloted – with very low direct operating costs, low noise and zero "tailpipe" emissions. The white paper [1] gave commercial expression to research by NASA scientists and other researchers who hypothesized that a new era of transformational vertical mobility could dawn with the advent of distributed electric propulsion, the autonomous flight technology and 5G communication networks. Vertical mobility services could provide an attractive solution for areas where merely increasing two-dimensional capacity would even make more complicate situation on ground traffic. The users of air taxis will not only experience a very time-efficient mode of travel, but also have a safe, enjoyable flight experience. In general [3] the assumed price range for on-demand air taxis is from 8 to 18 US dollars per minute, which is comparable to a premium taxi fare, considering the greater speed, shorter distance, and potentially higher load factor of more than one passenger on board.

Another crucial factor for the emerging urban air mobility market is the appropriate infrastructure: eVTOL landing sites, charging infrastructures and maintenance facilities are among the key enablers for successful operational business models. Furthermore, urban aircrafts require a safe and unobstructed landing zone that needs to be approved by the authorities. Hospital to hospital ferry flights originate and terminate on well-established helipads that would be early candidates for the new generation of eVTOL aircrafts. A robust 5G cellular network will also be imperative to enable communication among eVTOL aircraft, between eVTOLs and other flying objects, and between eVTOLs and control centers.

The idea of distributed electric propulsion is to replace the single complex rotor system – cyclic, collective, swashplate, transmissions, gearboxes, shafting, hydraulics, etc. – with multiple simple thrusters, and (ideally) an efficient wing for higher speed/long range cruise. Advocates also believe in the distributed electric thrusters (propellers, fans, etc.) which show huge benefits in terms of safety, emissions, noise and community acceptance.

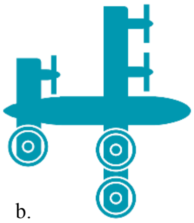
Electric propulsion will make eVTOLs cheaper than current models. Once the production of electric aircraft reaches maturity, the upfront cost of buying or leasing an eVTOL will be lower because electric powertrains are simpler than gas turbines. The costs of battery dropped in the previous years, thanks to the scale afforded by automotive manufacturers, and running an urban air taxi on electricity is more cost efficient than running a conventional helicopter on kerosene. Electric powertrains also have lower maintenance costs due to their simplicity, while technical services have to be able to deal with high voltage equipment. The total cost of ownership of eVTOL is therefore expected to be lower overall. Today available battery technology only allows short flight times for eVTOL applications.

It is yet to be determined what the first drone for urban air traffic will actually look like. The most promising architectures include multi- and quadrocopters, tilt-wingers, eVTOL aircrafts as well as hybrid constructions. While the wingless types are particularly suitable for inner-city operations in confined spaces, the winglet flying aircrafts are ideal for usage between cities or flying longer distances. Roland Berger [2] have done a study that compares the electric aircraft architectures that includes:



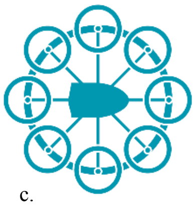
a.

Quadcopters. They represent wingless aircraft concept with four fixed propellers, possibly arranged as four sets of push-pull propulsion groups, can carry between 2 and 6 passengers at speeds of 120 to 150 [km/h] (Fig. 3a)



b.

Tilt-wing/convertible. These aircrafts have several propellers or ducted fans that can be tilted at different angles for fixed wings or tilting wings to achieve the different configurations needed for take-off, landing, flying and hovering. These aircrafts cater between 2 and 4 passengers and can reach speeds of 180 to 250 [km/h] (Fig. 3b).



c.

Highly distributed propulsion concept (multicopter) (Fig. 3c). This term designates wingless aircraft concept with more than four fixed propellers. These aircrafts cater to between 2 and 4 passengers and can reach maximum speeds of 80 to 100 [km/h].

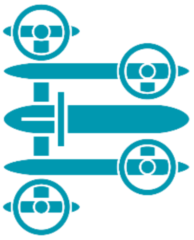
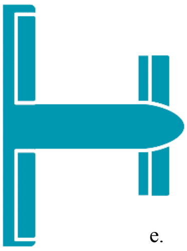


Fig.3d

Hybrid concepts. These concepts center around aircraft with fixed forward-facing propellers for forward movement and upward-facing/ retractable propellers to generate lift during the take-off and landing phases. Between 2 and 4 passengers can fly at speeds of 150 to 200 [km/h] in these vehicles (Fig. 3d).



e.

Fig.3. Roland Berger - Electric aircraft architectures [2]

Fixed-wing vectored thrust concepts. Winged VTOL jets are equipped with variable-direction fans. They too can accommodate 2 to 4 passengers and can fly at 200 to 300 [km/h] (Fig 3e).

Audi and Airbus presented a device concept named Pop.Up Next (Fig. 4) [11] that combines the flexibility of a ground vehicle with the freedom and speed of an eVTOL air vehicle, thus bridging the automotive and aerospace domains. The vehicle is made from three modules: base vehicle module, passenger capsule and quadcopter drone.

Considering the innovations in aircraft type and propulsion systems, we should put special attention on how to reduce the total weight of the air taxi vehicles. Therefore the key research objectives have to be taken into account: Propulsion efficiency – high power light battery, performance – aircraft drag minimization, operational effectiveness – costs, disturbance rejection, all-weather capability, noise and annoyance – aircraft arrangement, low tip speed, active noise control, structure and aeroelasticity – lightweight, durability, crashworthiness, structure health monitoring, wing-fuselage interaction over the conversion/ transition maneuver, aircraft design – handling qualities and vibration. The existence of light and efficient batteries is crucial, the current energy density is limited to 250 [Wh/kg], and a passenger drone can only be airborne for a maximum of 30 minutes [3], so the target will be more than 400 [Wh/kg]. Recently [12] announced that they possess a battery technology with energy density of 1000 [Wh/kg]. To succeed and build lightweight battery, probably it will have to be integrated together with the fuselage structure of the aircraft. It's really important to stay up to-date with the advancing composites manufacturers and 3D printing industry, as well as the correlation with the many different materials, equipment and technology suppliers, that want to share their experience and help develop future applications such an eVTOL aircrafts. On the other side we have advanced software tools as Computer Aided Design (CAD) and Finite Element Model (FEM) to design and co-simulate complex Multiphysics models,

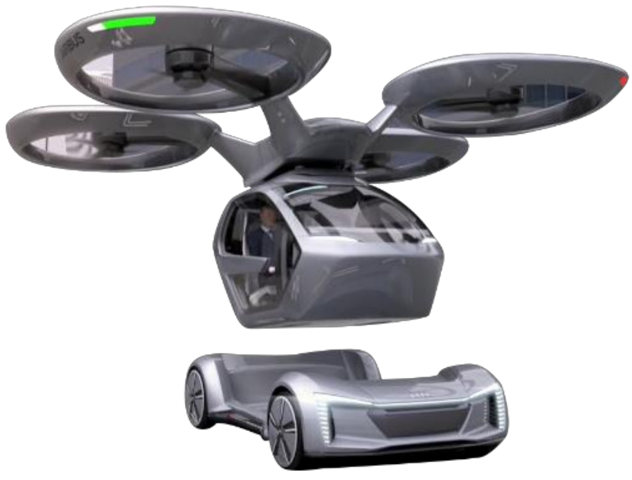


Fig.4. Audi and Airbus modular vehicle concept [11]

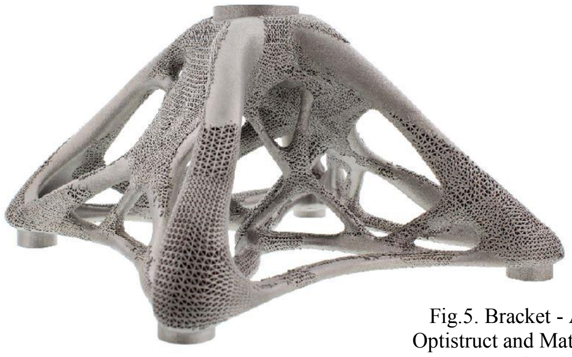


Fig.5. Bracket - Altair Optistruct and Materialise Magics software [13]

including coupled analyses with interaction of acoustics, thermal, electromagnetism, fluid and structure.

Fig.5 presents such a lightweight technology sample of a Titanium bracket [13] generated as a – hybrid lattice structure – combined with topological optimization. If we use such a printed part as an inset in to a Carbon Fiber Reinforced Plastic (CFRP) sandwich structural laminate, we can achieve a "super-structure", that will fulfill all of the requirements even for aero-flexible compliant structures. Fig.6 presents a profile of an Adaptive compliant wings, developed on the "Smart airfoil project" at ETH Zürich [14].

On Fig.7. National Aeronautics and Space Administration (NASA) [15] presents carbon fiber structural volumetric pixel (voxel). These pixels are modular units that can be arranged in repeating lattice-based space patterns. Connecting those voxels together in space, they built a structure of a large wing. During the flight it is possible to control and morph the wing with algorithms into the most efficient shape.

Considering the previously presented technology and advances in the material science, in order to build optimum lightweight structure probably we have to replace the traditional "passive" vehicle structure with a "smart" structure and gain weight over reducing the number of other system elements, for example the battery structure. In order to create this complex multifunctional main structure, the process will be expensive and obviously will increase the maintenance cost, due to the of reparation in case of damage. The structure health monitoring system must be incorporated in the real vehicle structure and control the aircraft loads during flight. In [16] additional, Safety of Flight Monitors or so-called Envelope and Floating limiter monitor was announced to address specific safety issues and concerns to down-mode the vehicle from "Advance" to "Safe"-mode (without IAS).

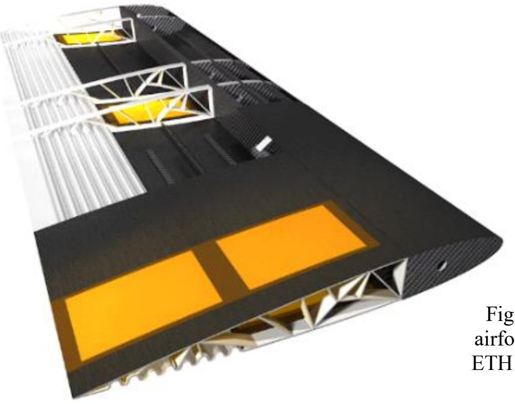


Fig.6. "Smart airfoil project" - ETH Zürich [14]

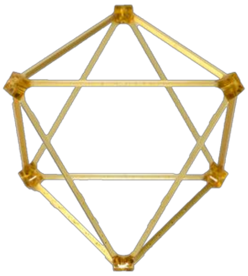


Fig.7. An individual modular block used for the MADCAT project - NASA [15]

It seems that it will take decades to transfer some volume-manufacturing techniques from automotive industry to aviation, and to integrate high-tech composites and alloys in this process will make it even more difficult to guarantee an eVTOL with low market price. Building fully composite aircraft like one from carbon fiber prepregs remains a largely hand-executed lamination work done from skilled workers.

As one could observe, the idea of vertical mobility is obvious solution to the traffic problems in the urban and non-urban areas. However, many issues need to be additionally solved in order to have safe and reliable vehicles. The aviation industry will have to develop completely new classes of aircraft with new flying abilities, using an efficient propulsion, flight control, and situational awareness.

V. CONCLUSIONS

Obviously, in the near future we should gain the experience of flying over the metropolitan cities sky. However, in order to achieve this goal, the researchers and the industry deal with many challenges in order to bring the personal flying industry to life. Probably the usage of electric vertical takeoff and landing (eVTOL) small aircrafts is the way to go, having argument that nothing else can compete it for efficiency [17], speed, reliability, safety, and quietness. Such a vehicle will not simply evolve from the existing hardware, either.

We have to walk a long way to see an operational commercial Intelligent Autopilot System (IAS), most likely a self-flying system. There will be at least two interim steps. The first step will involve a human pilot with extreme shortage of pilot's anticipation in to the system control and the second step will be the implementation of a fully autonomous flying system.

For private owned crafts, the expected scenario means that most of the systems either manage the act of flying or the processes of navigation and communication are controlled by the IAS, while the "pilot" or the owner who has limited or no flying skills, essentially is telling it what to do and where to go. In this scenario the human will decide and guide the craft over life critical conditions that are occurring irregularly and mostly stochastic, such as avoiding to flying over forest fire or sudden severe weather conditions not common in the region.

It seems that the battery technology advancing will not slow down, so it won't be limiting factor toward urban air mobility. Sooner or later the batteries will become lighter and more powerful [12]. However, the manufacturing will likely be a huge bottleneck. This is mostly because sustainable economics of an air taxi model massively depend on high volume aircraft manufacturing – or more thousand units per year. Most likely will take a decade to innovate more efficient volume composites lamination techniques suitable for the aviation industry.

REFERENCES

[1] “Fast-Forwarding to a Future of On-Demand Urban Air Transportation”, UBER Elevate, October 27, 2016.

https://www.uber.com/elevate/whitepaper

[2] “Urban Air Mobility - The rise of a new mode of transportation”, Roland Berger Focus, November 2018.

[3] “The Future of Vertical Mobility”, G.Grandl, M.Ostgathe, J. Cachay, S.Doppler, J. Salib, H. Ross, Porsche Consulting, 2018.

[4] National Highway Traffic Safety Administration. U.S. Department of Transportation, web media

https:/www.nhtsa.gov/technology-innovation/automated-vehicles-safety#issue-road-self-driving

[5] “How Tesla and Waymo are tackling a major problem for self-driving cars: Data”, Sean O’Kane, 04.2018

https://theverge.com/transportation/2018/4/19/17204044/tesla-waymo-self-driving-car-data-simulation

[6] Aptiv media web page.

https://www.aptiv.com/media/article/2018/01/07/the-autonomous-driving-platform-how-will-cars-actually-drive-themselves

[7] “Flight stability and automatic control. Vol. 2.” Nelson, Robert C. WCB/McGraw Hill, 1998.

[8] "An Intelligent Autopilot System that Learns Flight Emergency Procedures by Imitating Human Pilots", H. Baomar and P. J. Bentley, International Conference on Unmanned Aircraft Systems (ICUAS), Arlington, VA, USA, 2016.

[9] “Autonomous inverted helicopter flight via reinforcement learning”, Andrew Y. Ng, Adam Coates, Mark Diel, Varun Ganapathi, Jamie Schulte, Ben Tse, Eric Berger, and Eric Liang. Computer Science Department, Stanford University, Stanford, CA 94305. Whirled Air Helicopters, Menlo Park, CA 94025

[10] “Decision Making Under Uncertainty: Theory and Application, 1st Edition.” The MIT Press. Kochenderfer, M. J., Amato, C., Chowdhary, G., How, J. P., Reynolds, H. J. D., Thornton, J. R., Torres-Carrasquillo, P. A., U¨ re, N. K., Vian, J., 2015.

[11] “Conquering the Third Dimension” Volkswagen AG, web media 03.2019

https://www.volkswagenag.com/en/news/stories/2019/03/conquering-the-third-dimension.html

[12] Innolith AG, web media April 4, 2019

https://innolith.com/innolith-energy-technology-brings-1000km-ev-within-range/

[13] “The Spider Bracket: A Topology Optimization Project by Altair, Materialise and Renishaw”, web media

https://www.materialise.com/en/cases/spider-bracket-a-topology-optimization-project-by-altair-materialise-and-renishaw

[14] “Multidisciplinary Optimization of Morphing Wings with Distributed Compliance and Smart Actuation” - Composite Materials and Adaptive Structures Lab at Eidgenössische Technische Hochschule Zürich (ETH Zürich) 2009

http://www.structures.ethz.ch/research/aerospace-structures/structural-analysis-and-optimization.html

http://www.structures.ethz.ch/research/aerospace-structures/smart-airfoil.html

[15] “Mission Adaptive Digital Composite Aerostructure Technologies (MADCAT) Flexing Wings for Efficient Flight” - National Aeronautics and Space Administration (NASA), April 3, 2019

https://www.nasa.gov/feature/ames/madcat

[16] “Autonomous Vehicles” – Technology report, Woodside Capital Partners International, LLC, and Tracxn. November 2018.

[17] “Role of flying cars in sustainable mobility”, A. Kasliwal, N.J. Furbush, J.H. Gawron, J.R. McBride, T.J. Wallington, R.D. De Kleine, H. Chul Kim & G. A. Keoleian, April 2019

https://www.nature.com/articles/s41467-019-09426-0

Data-driven Framework for Enabling Adaptability in IoT-based Smart Grid Architecture

Nenad Petrovic
Faculty of Electronic Engineering,
University of Nis
Nis, Serbia
nenad.petrovic@elfak.ni.ac.rs

Djordje Kocic
Faculty of Electronic Engineering,
University of Nis
Nis, Serbia
seriousdjoka@gmail.com

***Abstract*—The increasing usage of electric power in recent years has introduced new challenges and requirements. The existing infrastructure and systems should become more flexible, able to operate in highly dynamic environments and react adequately to changes that occur. For that reason, the existing electric grid evolves towards Smart Grid architecture. In this paper, we explore how state-of-art information and communication technologies can be leveraged to enable adaptability in Smart Grid relying on affordable IoT devices. As an outcome, architecture based on the considered technology is proposed and several aspects of its implementation are presented in more details.**

Keywords—Edge computing, IoT, model-driven engineering, Smart Grid

I. Introduction

The utilization of electrical power has been the key-enabler for rapid technological progress of human society in previous century [1]. Nowadays, the demand for electrical energy is becoming higher, while the availability of non-renewable resources is limited, putting more pressure on the existing energy distribution systems, which are identified as a potential bottleneck [1, 2]. Such overload can cause serious problems and dramatically affect the quality of the transferred power. From the other side, despite the increased demand, consumers expect more affordable price of electrical energy. Therefore, there is a need for evolution of the existing energy distribution systems and increasing their flexibility [2]. The ability to operate in highly dynamic environments and adapt to the changes accordingly are seen as key features of modern systems and services [3].

For that reason, in recent years, a lot of effort has been put in transformation of the existing energy distribution systems in fusion with information and communication technologies towards the so-called *Smart Grid*. Smart Grid is defined as a next generation power grid, implemented as a two-way cyber-physical system with embedded computational intelligence, leveraging the collected information in order to provide clean, safe, secure, reliable, resilient, efficient, economic and sustainable electrical energy to end-users [1, 2, 4, 5]. One of its main characteristics is the ability to detect the events that occur anywhere in the grid and adopt the corresponding strategy in order to respond the changing demands or recover itself in case of anomalies in near real-time [1, 2, 4 ,5]. The implementation of Smart Grid is becoming of strategic importance for many countries, but its deployment requires large investments that could be very risky at the same time [1, 2, 4].

In this paper, we explore how the synergy of cutting-edge information and communication technologies can enable adaptability within Smart Grid relying on Internet of Things (IoT) devices and collected measurement data. As an outcome, we propose a Smart Grid architecture leveraging the mentioned concepts and present proof-of-concept implementation of some system aspects.

II. Background and Related Work

In this section, we first provide an overview of Smart Grid architecture model adopted from the existing literature. After that, we explore state-of-art information and communication technologies focusing on how they can be used in context of Smart Grids and what could be their possible contribution, taking into account the related work and use cases.

A. Smart Grid architecture

In literature, many descriptions of Smart Grid infrastructure components and architecture model exist [4-6]. However, some common elements are identified and we summarize them as follows: 1) *energy subsystem*, responsible for power generation, transmission and distribution 2) *communication subsystem,* that considers usage of wireless and wired communication technologies in order to enable the information exchange between Smart Grid components 3) *metering subsystem,* which consists of devices recording both electrical and non-electrical measurement values 4) *data layer*, that deals with concerns of representation and storage of the collected information – from sensor data and measurements to transactions 5) *computational intelligence subsystem*, which has goal to extract knowledge from the collected data and decisioning mechanisms that act according to that knowledge 6) *application layer*, that includes different kinds of software used either by Smart Grid operators and consumers providing a set of functions related to visualization, monitoring and control.

In this paper, the focus is on information and communication technology rather than components responsible for energy generation and distribution.

B. Internet of Things (IoT)

Internet of Things (IoT) refers to a system of various interconnected device used in everyday life with goal to perform the automation of a particular domain – from home appliances and product manufacturing to military systems [2, 6]. These devices are equipped with different kinds of sensors, cameras and positioning modules in order to collect certain type of information. Moreover, they can be equipped with actuators in order to be able to affect the environment. In most cases, their processing power is quite limited and they often need to communicate with other devices (or servers) in order to achieve their goal. For communication, a variety of technologies is used, both short- (such as Bluetooth) and long- range (Wi-Fi, 4G). At the moment, the next-generation 5G mobile networks are emerging, promising much higher bandwidth, capacity and reliability that will be highly beneficial for IoT use cases [7].

It is identified that IoT has great potential in Smart Grid applications [2, 6]. Smart Grids require measurement and actuation devices to be distributed throughout residential and industrial objects in order to collect the necessary data and provide the response to the events that have occurred. IoT devices perfectly fit that purpose, taking into account their small size, affordability and the fact that internet connection is a commodity nowadays.

C. Data analysis

In Smart Grids, it is necessary to analyze the enormous amount of data acquired by IoT and metering devices to extract knowledge and meaningful patterns in order to detect occurrence of particular events within the environment and react accordingly.

For that purpose, various data mining and machine learning techniques are used. In most of the existing work, the focus of their application is on anomaly detection and load forecasting [4]. In these use cases, clustering, classification, regression and association rule discovery algorithms are widely used, acting on various measurements collected by Smart Meters – from electrical signals to temperature, weather, rainfall and location data [4, 5, 6, 8, 9].

Anomaly detection is of utmost importance as it provides the ability discover malfunctions and failures, so Smart Grid can respond and fix them, making it self-healing [4, 5]. In [5], a clustering algorithm together with association rule discovery was applied in order to detect anomalous events, such as overcurrent. On the other side, to optimize the demand scheduling, the accurate energy usage pattern of the consumers is essential. This is where the demand forecasting has an important role. In [8] and [9], regression based on support vector machines was used for load forecasting.

D. Edge computing

Acquiring the data coming from IoT devices and their sensors is of utmost importance for monitoring and decisioning in Smart Grids. However, the traditional Cloud computing approach leads to unsatisfactory results, as offloading the enormous amount of data generated by IoT devices equipped with a variety of sensors to the Cloud for processing would introduce huge latency. On the other side, as Smart Grids grows, the number of connected IoT devices increases introducing even more delay.

As Smart Grid has to react to the environment changes and events in near real-time, such delay is intolerable [10, 11]. For that reason, Edge computing paradigm is considered as a possible solution. The idea of Edge computing is to move the computation and data processing closer to data sources – to the Edge servers, in order to enable faster response time [10]. For example, in [12], it has been shown that Smart Grid system monitoring performance can be increased up to 10 times by moving the computation closer to the location where the data was generated. Despite the significantly better performance of Edge computing-based architectures in case of monitoring, Cloud computing paradigm is still crucial when it comes to scenarios such as optimization across many grids and long-term data analysis [11].

E. Semantic technology

The role of semantic technology is to encode the meaning of data separately from its content and application code. This way, it is enabled that both machines and people can understand the data, exchange it and perform reasoning.

In context of semantic technologies, ontologies are used to describe the shared conceptualization of a particular domain. Semantic descriptions are stored within the triple stores. RDF[1] is often used for the representation of semantic data within triple stores. It consists of classes, their properties and relationships expressed in forms of triplets (*subject, predicate, object*). SPARQL[2] is a language used for querying the RDF semantic triple stores. By executing queries against the triple store, it is possible to retrieve the results that can be further used by reasoning mechanisms in order to infer new knowledge based on the existing facts.

In IoT systems, semantic technology is used for various purposes. It is widely adopted in cases when it is needed to achieve interoperability of heterogeneous devices [13]. Furthermore, the semantic technology is also used to support more sophisticated scenarios, such as providing means for automated code generation in process of device coordination [14, 15]. In [16], a lightweight semantic framework was used for semantic annotation of the results obtained by computer vision and audio analysis algorithms in order to enable reasoning about the events that occurred within IoT-based video surveillance system and act accordingly. In this paper, we want to adapt the similar approach to [16] for control within Smart Grid architecture, while relying on general ideas of [14] and [15], but considering the process of adaptation instead of coordination.

F. Domain-specific languages and modelling notations

Domain-specific language (DSL) is a programming language specialized for solving problems from a particular application domain. If the considered problem belongs to target domain, that problem is solved more conveniently than using general purpose programming languages, but it might not expressive enough to solve problems outside that domain. Their notation could be textual or visual (within modeling tools). Domain-specific languages are being adopted in IoT systems in order to decrease cognitive load introduced by the heterogeneity of devices and complexity of the underlying infrastructure. The domain-specific language scripts are further automatically translated to lower-level device-specific commands.

For example, in [17], Experiment Description Language (EDL), a domain specific language was used to describe the experiments carried out using robotic IoT devices. In this paper, the idea is to leverage domain-specific notation to enable modelling of systems aspects related to adaptability. A closely related existing solution is Cloud Application Modelling and Execution Language (CAMEL) [18], which enables the specification of multiple aspects of cross-cloud applications, providing the means for modelling of adaptive behavior as a reaction to various events related to change of Quality of Service parameters.

In context of Smart Grids, visual tools based on domain-specific languages would enable much more convenient control and management for operators. This way, the implementation of complex scenarios involving adaptability

[1] https://www.w3.org/RDF/

[2] https://www.w3.org/TR/sparql11-query/

is enabled by eliminating the need to know in-depth implementation details of the involved devices within the Smart Grid system.

III. Implementation Overview

In this section, we propose a data-driven Smart Grid architecture based on IoT devices by putting together the previously mentioned technologies. Moreover, some aspects of system implementation are described in details.

Architecture and working principle: The measurement of electrical and other useful quantities is performed by IoT and smart devices, often referred to as *Smart Meters*. They can either be microcontrollers, low-power single-board computers (such as Raspberry Pi) or even smartphones, as in our case. The advantage of using smartphones is the availability of built-in sensors and input ports (such as audio jack). When it comes to smartphones, their rechargeable batteries and wireless mobile network availability enable usage even outdoors and in less accessible areas. Moreover, the collected data is analyzed relying on data mining and machine learning techniques. The obtained results are semantically annotated, so the semantic reasoning can be leveraged to draw conclusions about the events that occurred within the Smart Grid environment. According to these results, the corresponding actions are taken in order to adapt the Smart Grid to current consumption demands and respond these events. A set of actions that has to be taken when specified condition is fulfilled (referred to as *adaptation strategy*) can be specified by operators, using a visual modeling tool relying on a domain-specific notation. Finally, the device-specific commands are generated in order to respond to the changes detected in Smart Grid. The illustration of working principle is given in Fig. 1.

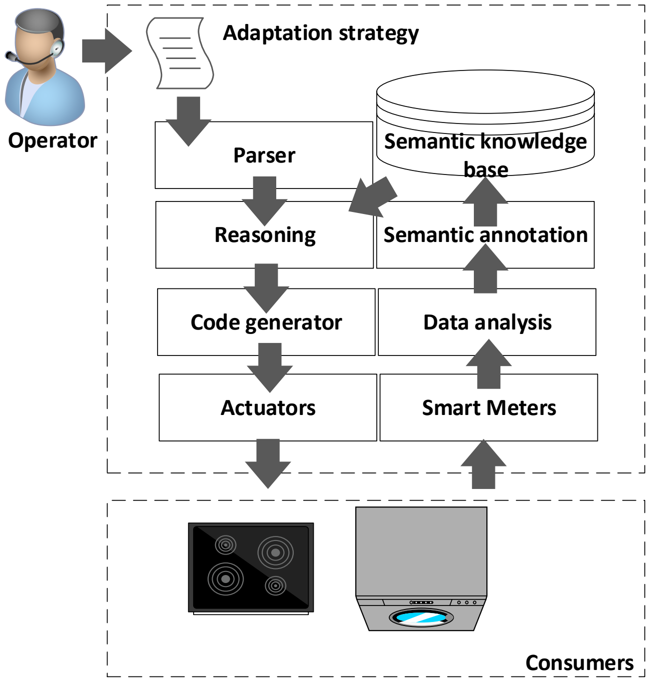


Fig. 1. Data-driven framework for enabling adaptability of IoT-based Smart Grid architecture

Android smartphones as Smart Meters: In [19], we have presented a method for acquiring electric measurement utilizing affordable Andorid-based devices. Voltage and current signals are acquired via voltage and current transformers from the power grid. Both signals are converted into voltage signals, which are then scaled down further to audio signal levels using variable resistors. After that signal goes directly into the devices 3.5 mm audio jack. Since many Android based devices support stereo microphone input, that makes an ideal 2 channel measuring platform for power signals. Devices audio system (sound card) performs analog to digital A/D conversion, and the data can now be processed by standard digital signal processing methods, such as FFT-based algorithms. Moreover, it is convenient to also record other data coming from device sensors, such as temperature and location, which can be used for more sophisticated data analysis techniques, especially when it comes to load forecasting. The collected data is sent to the Edge server each second via Wi-Fi or mobile network. The smart measurement system is illustrated in Fig. 2.

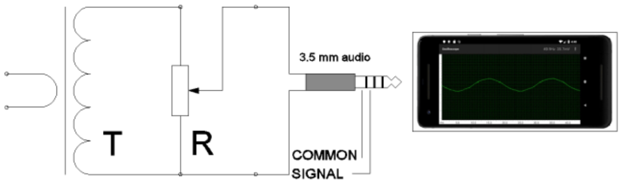


Fig. 2. Android based smart measurement system

However, using smart devices based on Android platform certainly has some down sides. Biggest disadvantage is that sound card systems can only detect periodic signals, so it is not able to register DC offset. Another disadvantage is that integrated stereo input supports only 2 channels, which does not pose a problem if external multichannel sound card is used. Lastly, smart devices have limited processing capabilities, but software optimization would overcome that disadvantage. Since Android devices are based on Linux kernel that is not a hard real-time system, it would not be suitable for high reliability applications, but it is exceptional for measurement and support uses.

Communication protocol: For machine-to-machine communication (IoT devices and Edge servers), we decided to use MQTT (Messsage Queuing Telemetry Transport)[3]. It is a lightweight, publish-subscribe-based ISO-standard messaging protocol, working on top of TCP/IP. MQTT is designed for use cases where small code footprint is desired or network bandwidth is limited, which suits well the scenario of IoT-based Smart Grid. However, publish-subscribe messaging mechanism requires a message broker. For that purpose, we use a Node.js MQTT broker implementation within Node-RED[4], deployed on Edge server. For Android client implementation, Paho Android Service[5] was used. The devices measuring the electric signal send messages via MQTT protocol to the Edge broker that is responsible for their delivery to all the topic subscribers. Messages are sent as JSON-encoded string, such as the following used for

[3] http://mqtt.org/

[4] https://flows.nodered.org/node/node-red-contrib-mqtt-broker

[5] https://github.com/eclipse/paho.mqtt.android

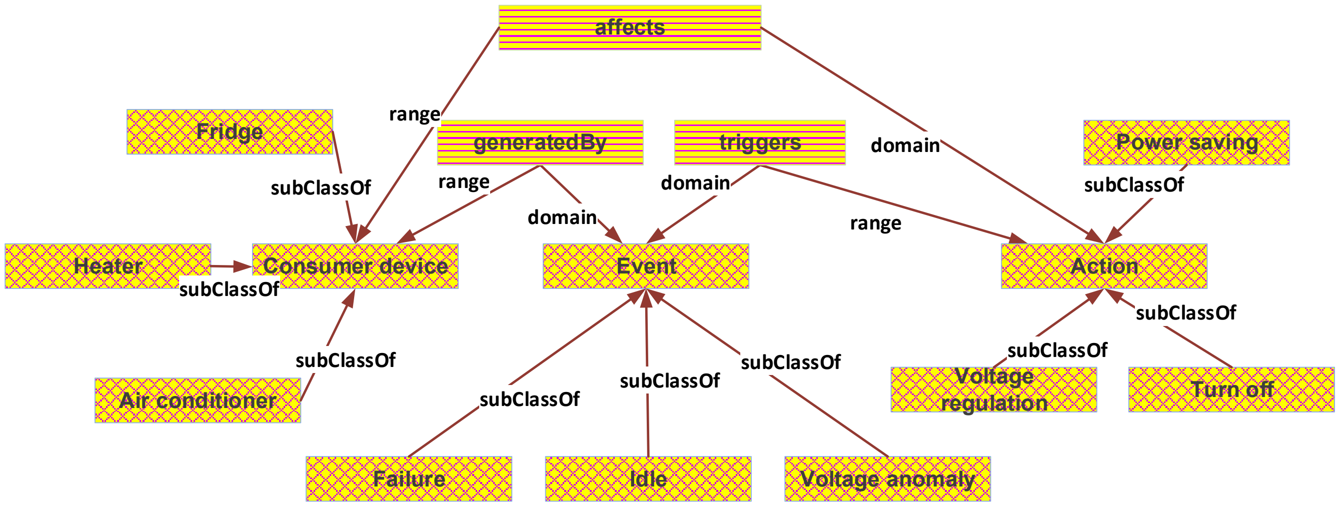


Fig. 3. Domain ontology for Smart Grid control

measurement of electrical signal frequency: `{"device":"AndroidSmarthpone1", "sensorType":"frequency", "value":49.7}`. On the other side, MQTT messages are also used to send command parameters to actuator devices.

Data analysis module: For implementation of data analysis mechanisms, we rely on Java-ML[6] [20] library for Java programming language that offers a simple API to data mining and machine learning algorithms, such as clustering, classification and regression. In this paper, we focus on anomaly detection based on *k*-nearest neighbor (*k*-NN) classification algorithm [21] leveraging voltage and frequency measurement data. Classification is process of identifying to which set of categories a new observation belongs, on the basis of a training set of data containing observations (or instances) whose category membership is already known. The k-NN algorithm tries to classify an unknown sample based on the known classification of its *k* neighbors (surrounding samples). In context of anomaly detection, beside the values of electrical signal measurement, the training data set also contains a label with two possible values: 1) "ok" - in case of normal consumer device operation 2) "anomaly" – in case of malfunction. Voltage and frequency are considered as independent, while the anomaly label is a dependent variable. The task of classification algorithm is to assign a correct label to each new measurement from the test set.

Semantic framework: A domain ontology (illustrated in Fig. 3) is defined with purpose to semantically annotate the results obtained as output of data analysis module. This way, it is enabled to draw conclusion about the events that occurred by executing SPARQL queries and interpreting their results. Different types of events are considered, such as voltage anomaly and device failure. Furthermore, for each of the available events, we define possible actions to be taken in order to adapt to detected events, such as voltage regulation, turning off the device, switching the device to power saving mode. In Listing I, an example of SPARQL query for voltage anomaly detection is given.

LISTING I. SPARQL QUERY EXECUTED IN CASE OF VOLTAGE ANOMALY DETECTION

```
PREFIX scg: <http://www.example.com/resources/scg/>
SELECT ?Event
WHERE {
  GRAPH <http://www.example.com/test1> {
  ?Event scg:detectedBy ?Device.
          FILTER(regex(STR(?Event), "voltage-anomaly"))
  }
}
```

Visual modelling tool for operators: A visual modelling tool for Smart Grid operators was developed using Node-RED as a basis. The domain-specific notation within the tool is described by a metamodel shown in Fig 4. It is a model of a modeling language which defines the structure and constraints for a family of models. In this paper, it is used to define a set of actions that need to be taken over the target devices in order to adapt Smart Grid to the environment changes when pre-defined environment conditions are satisfied. The conditions could be either some specific events or relational expressions with respect to a given threshold.

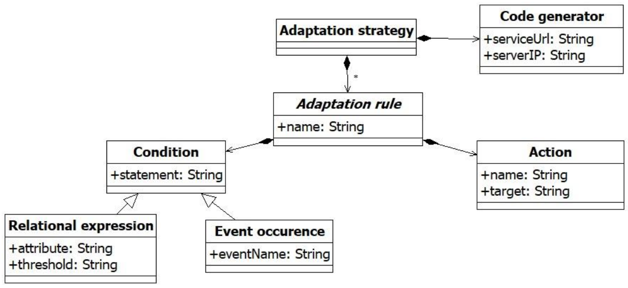
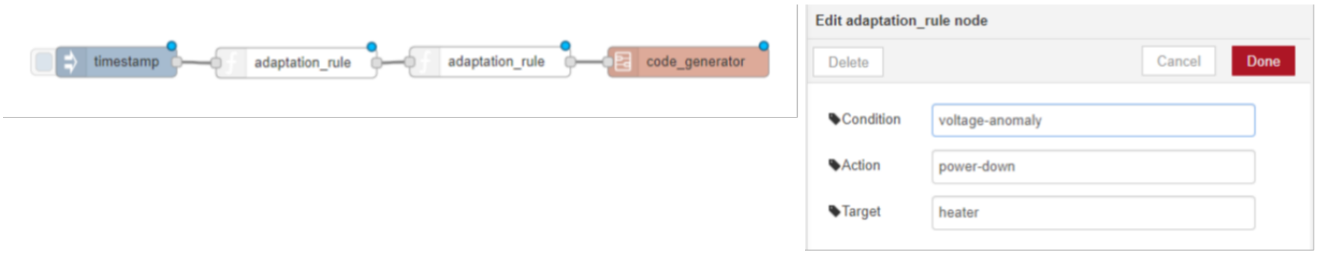


Fig. 4. Adaptation strategy metamodel given in UML notation

Node-RED[7] is an intuitive and extendable framework that is used for wiring together IoT devices, APIs and online services in novel ways, providing a browser-based editor with drag-and-drop user interface using the wide range of modeling elements (nodes). Therefore, in order to enable adaptation strategy definition within Node-RED, we had to develop custom node elements to embed the described domain-specific notation. A similar tool was used for implementation

[6] http://java-ml.sourceforge.net/content/cite-java-ml

[7] https://nodered.org/



Condition: voltage-anomaly
Action: power-down
Target: heater

Fig. 5. Adaptation strategy visual modelling tool based on Node-RED

of coordination flow modeling in case of unmanned vehicle coordination [22].

Two type of custom nodes are imported into Node-RED environment: *adaptation_rule* and *code_generator*. Adaptation rule defines (*condition->action*) mapping expressing the fact that when condition is fulfilled, the corresponding action will be triggered to change the state of a target device. On the other side, Code generator is an element that provides parameters needed to invoke the code generation on Edge server. This way, the modelling tool remains independent from the implementation of backend and command generation mechanisms. In Fig. 5, an example of adaptation rule in Node-RED that defines that a heater device has to be turned off in case of voltage anomaly is given.

Reasoning, command generation and execution: Command generation is performed according to the algorithm shown in Listing II (inspired by the algorithm presented in [22]).

LISTING II. COMMAND GENERATION ALGORITHM PSEUDO-CODE

```
Input:  measurements, adaptation strategy
Output:  device-specific command script
Steps:
  1. Retrieve all the adaptation rules from the adaptation strategy;
  2. Analyze measured data;
  3. Semantic annotation of obtained results;
  4. For each adaptation rule
  5.     If(adaptation_rule.condition is true)
  6.         then generate command targeting adaptation rule.target;
  7. end for each
  8. end
```

First, the measured data is analyzed using the available data mining and machine learning algorithms. Moreover, the obtained results are semantically annotated according to the domain ontology. In semantic reasoning phase, each condition from the adaptation strategy is translated to SPARQL query and executed against the semantic knowledge base. If that condition is fulfilled, then the corresponding device-specific code is generated in order implement the selected action for the target device and appended to the command script.

Once submitted for execution, Edge server is responsible for the delivery of command parameters via MQTT protocol to the actuators responsible for control of target devices.

IV. RESULTS AND EVALUATION

Experimental results achieved utilizing the framework presented in this paper are considered from two different points of view. First, we consider the accuracy of classification algorithm applied for short-term detection of anomalies based on electrical measurement data. Moreover, the responsiveness of the implemented system is analyzed taking into account the processing time necessary for specific steps. The evaluation was performed on a laptop equipped with Intel i7 7700-HQ quad core CPU running at 2.80GHz and 16GB of DDR4 RAM.

A. Anomaly detection based on classification

In Table I, the achieved results using the presented approach for anomaly detection based on classification algorithm implemented using Java-ML are given. The first column shows the experiment number. The second column represents the number of measurements used for classification algorithm training. The third column shows the size of test set, while the last column shows the achieved accuracy expressed as ratio between correctly classified and total number of test samples (shown as percentage).

TABLE I. CLASSIFICATION FOR ANOMALY DETECTION

Case	*Training set size*	*Test set size*	*Correct/Test set size [%]*
1	50	180	87,2
2	100	180	93,3
3	135	180	96

As it can be seen in the results, the anomaly detection mechanism based on classification gives satisfactory results in cases of different training/test set ratios, but performs better in case of larger training set. The value of *k* parameter used in *k*-NN algorithm was 5 in our case, as it led to the greatest percentage of correctly classified observations.

B. Processing time

In this subsection, we present performance evaluation considering the various steps necessary to be taken in order to detect anomalies within Smart Grid and react accordingly.

In Table II, an overview of the achieved processing time in each step is shown for various cases of adaptation strategy length and consumer devices involved. For each case the following processing times (given in seconds) are considered: data analysis time, SPARQL queries and command generation. The times shown are calculated as average of 10 experiments.

As it can be seen, the data analysis time depends on number of devices involved, as more devices generate larger amount of data that has to be analysed. In our experiments, each rule targeted a distinct device, while data analysis was performed for measurements collected during 120 seconds.

Moreover, the time needed for SPARQL query execution depends on number of adaptation rules involved, as each adaptation rule contains a condition element that is translated to a query.

Finally, command generation time depends on number of adaptation rule conditions that are true, as commands will be generated only in these cases. Therefore, indirectly, it also depends on number of adaptation rules.

TABLE II. PROCESSING TIME EVALUATION RESULTS

Number of rules	**Number of true cond.**	***Data analysis [s]***	***SPARQL queries [s]***	***Command generation [s]***
1	1	2.14	1.16	0.97
2	2	3.89	1.78	1.76
2	1	3.81	1.72	0.89
3	3	6.01	2.19	2.31

V. CONCLUSION AND FUTURE WORK

In this paper, the enabler technologies for data-driven IoT-based Smart Grid architecture were discussed and some implementation aspects presented. It can be concluded that IoT-based technology has huge potential in this use case. The implemented anomaly detection mechanism gives satisfactory results, while the achieved processing time in given configuration is acceptable even for use in near real-time.

However, our plan is to further work on the implementation, focusing on optimization of real-time performance and security issues. Security is of utmost importance in IoT-based Smart Grid architecture, as device communication is often performed via vulnerable wireless channels. In this case, attacks can lead to catastrophic consequences, such as nation-wide power outage and physical damage to infrastructure and beings [23].

Moreover, additional data analysis techniques will be adopted, such as consumption forecasting, giving the ability to make Smart Grids even more adaptable to consumer demands. The performances of various algorithms used for anomaly detection and load forecasting will be compared in order to find the right choice, both in terms of accuracy and minimal processing time required.

ACKNOWLEDGMENT

This work has received funding from the European Union's Horizon 2020 Framework Programme for Research and Innovation under the Grant Agreement No 645220, project RAWFIE (Road-, Air- and Water- based Future Internet Experimentation) and is partially supported by grants III44006 and III47003 by the Ministry of Education, Science and Technological Development of the Republic of Serbia.

REFERENCES

[1] A. H. Bagdadee, L. Zhang, Smart Grid: A Brief Assessment of the Smart Grid Technologies for Modern Power System, Journal of Engineering Technology, vol. 8, no. 1, January 2019, pp. 122-142, 2019.

[2] M. Jaradat et al., "The Internet of Energy: Smart Sensor Networks and Big Data Management for Smart Grid", Procedia Computer Science 56 (2015) , pp. 592–597, 2015.

[3] R. Kazhamiakin, S. Benbernou, L. Baresi, P. Plebani, M. Uhlig, O. Barais, "Adaptation of Service-Based Systems", Service Research Challenges and Solutions for the Future Internet, Lecture Notes in Computer Science, vol 6500. Springer, Berlin, Heidelberg, pp. 117-156, 2010.

[4] B. Rossi, S. Chren, "Smart Grids Data Analysis: A Systematic Mapping Study" [submitted for publication], pp. 1-26, 2018. Available on: https://arxiv.org/pdf/1808.00156.pdf

[5] B. Rossi, S. Chren, B. Buhnova, T. Pitner, "Anomaly Detection in Smart Grid Data: An Experience Report", IEEE International Conference on Systems, Man, and Cybernetics (SMC 2016), pp. 1-6, 2016.

[6] H. Shahinzadeh, J. Moradi, G. B. Gharehpetian, H. Nafisi, M. Abedi, "IoT Architecture for Smart Grids", 2019 International Conference on Protection and Automation of Power System (IPAPS), pp. 22-30, 2019.

[7] W. Ejaz et al., "Internet of Things (IoT) in 5G wireless communications", IEEE Access 4, pp. 10310–10314, 2016.

[8] M. Božić, M. Stojanović, Z. Stajić, "Short-Term Electric Load Forecasting Using Least Square Support Vector Machines", Facta Universitatis, Series: Automatic Control and Robotics vol. 9, no. 1, pp. 141-150, 2010.

[9] A. B. M. S. Ali and S. Azad, "Demand Forecasting in Smart Grid"" Green Energy and Technology, pp. 135–150. doi:10.1007/978-1-4471-5210-1_6

[10] W.Z. Khan et al., Edge computing: A survey, Future Generation Computer Systems (2019), pp. 1-44, 2019.

[11] F. Samie, L. Bauer and J. Henkel, "Edge Computing for Smart Grid: An Overview on Architectures and Solutions", Power Systems, pp. 21–42, 2018.

[12] Y. Huang et al., "An Edge Computing Framework for Real-Time Monitoring in Smart Grid", 2018 IEEE International Conference on Industrial Internet (ICII), pp. 99-108, 2018.

[13] R. Agarwal et al., "Unified IoT ontology to enable interoperability and federation of testbeds", 2016 IEEE 3rd World Forum on Internet of Things (WF-IoT), pp. 70-75.

[14] V. Nejkovic, N. Petrovic, N. Milosevic, M. Tosic, "The SCOR Ontologies Framework for Robotics Testbed", 2018 26th Telecommunication Forum (TELFOR), Belgrade, pp. 1-4, 2018.

[15] N. Petrovic, V. Nejkovic, N. Milosevic, M. Tosic, "A Semantic Framework for Design-Time RIoT Device Mission Coordination", 2018 26th Telecommunication Forum (TELFOR), Belgrade, pp. 1-4, 2018.

[16] N. Petrovic, "Surveillance System Based on Semantic Video and Audio Annotation Leveraging the Computing Power within the Edge, XIV International SAUM 2018, pp. 281-284, 2018.

[17] K. Kolomvatsos, M. Tsiroukis, S. Hadjiefthymiades, "An Experiment Description Language for Supporting Mobile IoT Applications", FIRE Book, European Commission, River Publishers, 2016, pp. 461-486.

[18] A. Rossini et al., "The cloud application modelling and execution language (CAMEL)," Open Access Repositorium der Universität Ulm, pp. 1-39, 2017.

[19] Đ. Kocić, N. Petrović, "Application of Android Based Devices in Analog Electric Signal Measurement", YuInfo 2019, Kopaonik, Serbia, pp. 1-5, 2019.

[20] T. Abeel, Y. V. de Peer, Y. Saeys, "Java-ML: A Machine Learning Library", Journal of Machine Learning Research, vol. 10, pp. 931-934, 2009.

[21] T. Cover and P. Hart, "Nearest neighbor pattern classification", IEEE Transactions on Information Theory, 13(1), pp. 21–27, 1967.

[22] N. Petrovic, M. Tosic, V. Nejkovic, N. Milosevic, "Formalizing Device Coordination in IoT Systems: The SCOR Case Study", YuInfo 2019, pp. 1-6, 2019.

[23] V. Delgado-Gomes, J. F. Martins, C. Lima, C., P. N. Borza, "Smart grid security issues", 2015 9th International Conference on Compatibility and Power Electronics, pp. 534-538, 2015.

Extracting semantic information from text for generating visual 3D scenes

Aleksandar Despotovski
Faculty of Computer Science and Engineering
Ss. Cyril and Methodius University
Skopje, North Macedonia
aleksandar.despotovski@students.finki.ukim.mk

Sonja Gievska
Faculty of Computer Science and Engineering
Ss. Cyril and Methodius University
Skopje, North Macedonia
sonja.gievska@finki.ukim.mk

***Abstract*—This research explores the feasibility of generating a 3D scene that depicts a given text by extracting semantic information from the narrative. For a particular text created by a user, the system presented in this paper extracts object which could be visualized, their attributes and the spatial relationships between them. Additional explicit properties on the objects that are learned from a 3D scene dataset are utilized as well.**

***Index Terms*—nlp, information extraction, grammar parsing, 3d scenes**

I. Introduction

Humans are visual creatures that tend to remember scenes much better than words. Hence, the old saying was born: ”A picture is worth a thousand words”. In many situations, we find ourselves struggling to keep up with complex ideas, where a single picture would explain them perfectly. Take for example, the intuition behind Pythagoras' theorem: if you don't see the squares put next to each other to form a triangle between them visually, you will have a hard time wrapping your head around it. Other times, having pictures are a necessity. If you have an idea for new furniture for your home, you'll have to show the carpenter a rough sketch of it. However, generating visualizations is a hard task; one would need some expertise in order to generate decent-quality 2D or 3D models, not to mention it's very time-consuming. An easier method for translating people's thoughts into visual models would be of great benefit in many areas, including architecture, game design, improvement of computer-aided design software, and even entertainment. The goal of this project is to provide a way of extracting the semantic information from the text that's needed for generating a plausible 3D scene that's closest to the user's thoughts.

There are many challenges in transforming the user's sentence into a representation that's comprehensible for the computer. The first one that comes to mind is the ambiguity of human's languages. There are countless of words that change their meaning based on context and use in sentence, and we can't hope to grasp their intended meaning at all times. There are also many ways to express one's thought, so we can try to parse at least some of the most useful sentence patterns in our domain. Another problem arises with the visualization of the scenes: what means for an object to be *to the left of* another object, or *on top of* another one? Or a person to *have* a hat? What difference would it make if we say a person *has* a pen? Well, we would obviously draw the person with the pen in his hand, but it would be arguably more precise to draw him with a hat on his head.

In this paper we try to solve these problems and provide a representation for the visualizable objects parsed from text, as well as the spatial relations between them.

II. Related work

In recent years, numerous articles and papers have been published on this topic, and a few visual scene generating systems have been developed. Most notably, Coyne et al. of AT&T Labs Research have developed WordsEye text-to-scene conversion system [1]. Its purpose is the same we are trying go achieve with our paper: given a textual description of a scene, generate the most plausible 3D model representing it. WordsEye first parses and tags the input text, and then converts it to a dependency structure, which is subsequently semantically interpreted and translated into semantic representation of the user's text. After that, the semantic representation is converted to 3D objects, relations and attributes according to some depiction rules. Chang et al. have done something similar with SceneSeer [2], which is also a tool for generating 3D scenes with natural language. It is able to create scenes according to previously learned ones, and allows interaction with the scene, for example, moving placed objects, changing the camera position, etc. This work is based on their previous research [3], which gives the basic steps needed to achieve their system.

III. System description

The main problem this paper focuses on is extracting the meaningful data that can be used for generating a 3D scene from sentences written in English, and generating a machine representation of the visualizable objects that could later be used to efficiently draw them as a part of a scene. The second problem we tackle is to find the most suitable 3D model for each object on the scene from a 3D model dataset. The dataset we use in this project is Stanford's SceneSynth database of 3D models and scenes [4].

With that being said, the system is split into two logical components:

- *Sentence parser*, which extracts the semantic information from the sentences,

- *Model finder*, which matches the discovered objects by the sentence parser with the most probable model representations.

A. *Object representation*

Each object is described by its name, attributes and relations to other objects. The objects are analogous to noun phrases in a sentence, and we track the nouns in the sentence to catch the drawable objects, as we'll see later. The attributes describe the objects, and correspond to the adjectives in a sentence. We divide the attributes in two categories: visualizable and nonvisualizable. The visualizable attributes are those that visually affect the object. Our system currently supports three types of visualizable attributes: count, color and shape. The count attribute is limited to the numbers from one to ten, while the shape attribute accepts the following values: *small, tiny, little, big, large, huge, humongous* and *bombastic*. The color attribute supports all colors available in the Matplotlib Python library [5]. The non-visualizable attributes provide better description of the objects, which is valuable when matching the items with the most probable 3D model representation of them. Every adjective that doesn't fall into the visualizable attributes is a non-visualizable attribute.

The relations are divided into functional (positional) relations and descriptive relations. The functional relations express the positioning of the objects relative to each other. The supported functional relations are *next to, near, to the left of, to the right of, on top of, below, above, in front of, to the back of, behind, with* and *have*. The descriptive relations connect the objects to attributes. The only currently supported descriptive relation is *be*. Figure 1 shows a diagram of a dependency structure that would be created if the user entered the corresponding sentences.

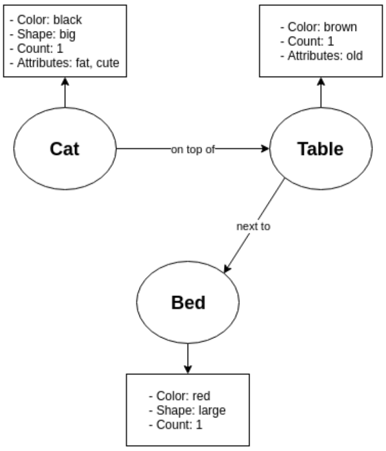


Fig. 1: The objects and relations extracted from the sentences: *"There is a big, black cat on top of an old, brown table. The cat is cute and fat. The table is next to a large red bed"*.

B. *Sentence parser*

The responsibility of the sentence parser is to extract the objects that should be drawn on the scene, their attributes, and the relations between them. The sentences are first tokenized. Then, the tokenized sentence is passed to a context-free grammar parser, from which it generates a parse tree. The generated parse tree is then annotated and evaluated, and the evaluation returns the detected objects along with their relations.

The sentence parser contains a *context*, which stores the currently detected objects, and is also used by the annotated tree evaluator for keeping track of the relevant objects for detecting relations and attributes. Figure 2 depicts the architecture of the sentence parser.

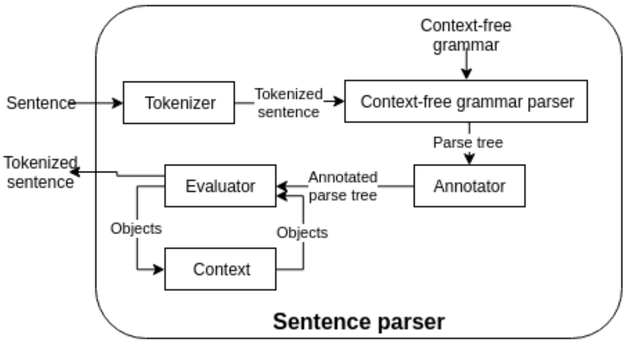


Fig. 2: A diagram describing the structure of the sentence parser

1) *Tokenizer:*

The tokenizer transforms the sentence into tokens that are meaningful to the context-free grammar parser. Firstly, the tokenization is performed with Spacy [6], a free NLP library for Python. Spacy extracts the lemma, simple part-of-speech (POS) tag and detailed POS tag for each token, and also finds the syntactic dependencies between tokens, which we don't need. Then, each token is transformed according to the following rules:

1) If the token is a punctuation, delete the token
2) If token's lemma is a *reserved* word and the token is not a determiner (*"the"*), set the token equal to its lemma
3) If token is a noun, set the token equal to *"NN"*
4) If the token is a verb, set the token equal to *"VB"*
5) If all else fails, set the token equal to its detailed POS tag.

The second rule mentioned *reserved* words. A *reserved* word is a word that appears in any of the relations. The reason why we don't want to appoint the lemma to a determiner (*"the"*) is because we want to catch it for several purposes in the grammar. The third rule just aggregates all nouns, personal and non-personal, into a single token. The fourth rule has similar meaning, but for verbs.

The return result from the tokenizer is a transformed tokenized sentence according to the rules above.

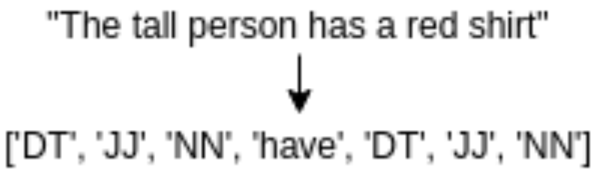

"The tall person has a red shirt"
['DT', 'JJ', 'NN', 'have', 'DT', 'JJ', 'NN']

Fig. 3: The result from the tokenization of the sentence ”The tall person has a red shirt”

2) Context-free grammar parser: The context-free grammar parser is defined by a context-free grammar, and accepts tokenized sentences as input, for which it generates a parse tree. The parser we use is the recursive descent parser of the NLTK library [7].

Our grammar consists of terminals, which include the sentence POS tags and the reserved words (defined in section 3.2.1), and non-terminals, which represent the logical substructures of the sentence.

The grammar currently supports two general types of sentences:

- Declarative (There is an old tree)
- Descriptive (The tree has green leaves)

Below are given the top-most sentence structure rules for our grammar:

- S $\rightarrow$ SENT
- SENT $\rightarrow$ ϵ
- SENT $\rightarrow$ THEREIS NPLIST RELLIST
- SENT $\rightarrow$ THEREIS NPLIST RELLIST ’CC’ SENT
- SENT $\rightarrow$ THEREIS NPLIST RELLIST ’IN’ SENT
- SENT $\rightarrow$ NPLIST RELLIST
- SENT $\rightarrow$ NPLIST RELLIST ’CC’ SENT
- SENT $\rightarrow$ NPLIST RELLIST ’IN’ SENT

Humans often express their thoughts informally, and don’t strictly follow the grammar rules of their language. Our context-free grammar is build with that in mind. It allows omissions of some words in certain sentences that aren’t of high importance. For example, the sentence *”There is a car to the left of the blue truck”* yields the same dependency structure as the sentence *”There is car to the left of blue truck”*.

NLTK’s parser mandates that the root non-terminal is always *’S’* and doesn’t appear on the right side in a rule. Because we use the recursive descent parser, our grammar doesn’t have left recursion, as then it would be useless. The tokens inside apostrophes are terminals, while the other are non-terminals. What follows is a brief description of the terminals and non-terminals that are given in the extract above:

- **S**: root non-terminal
- **SENT**: non-terminal that describes the structure of the sentences
- **NPLIST**: non-terminal that depicts a list of noun phrases. A noun phrase is a noun, possibly preceded by a determiner, a cardinal or adjectives
- **RELLIST**: non-terminal that represents a list of relations
- **THEREIS**: non-terminal that is made of the terminals ’EX’ ’be’
- **’CC’**: a conjunction terminal, such as *’and’*
- **’IN’**: a conjunction, subordinating or preposition

The result tree has non-terminal nodes that have the non-terminal labels, and children that are either terminal strings, or other non-terminal nodes.

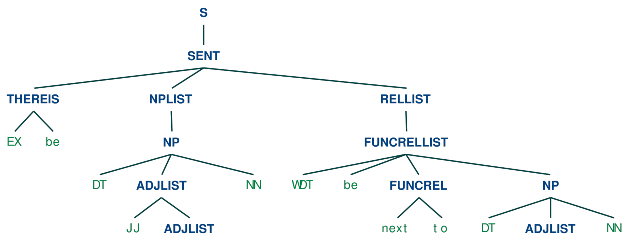


Fig. 4: The parse tree generated for the sentence ”There is a blue table which is next to the refrigerator”

3) Annotator: The parse tree generated by the parser is not very useful by itself, because it only gives a structure to the sentence. We need something more useful, a way to identify what to do at every node of the tree. For that reason, we have the tree annotator, which creates a new tree with the same structure as the parse tree, but with specialized nodes for each distinct non-terminal and terminal of the grammar.

The nodes of the annotated tree have evaluation logic and variables that are useful to them. With their help, we are able to extract the meaning of the sentences.

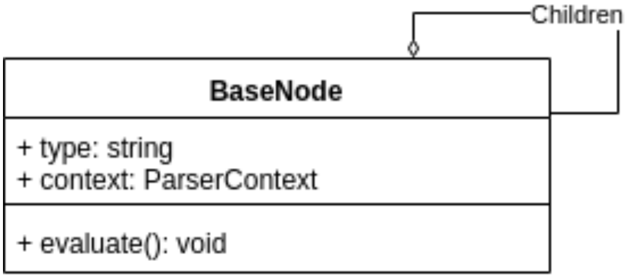


Fig. 5: UML class diagram for the base annotated tree node

As shown in Figure 5, the annotated tree nodes have a reference to the parser context. Its usage will be explained shortly.

4) Evaluator: After the annotated parse tree is created, it is passed to the evaluator, which evaluates it. The evaluation is recursive, it starts with calling root node’s *evaluate* method, which evaluates all of its children. Of course, each specialized tree node has its own evaluation logic. For instance, the *NPLIST* node’s evaluation comes down to recursively building the list of noun phrases. Some of the nodes have trickier tasks. For instance, the noun phrase node (*NP*) needs to find all the adjectives of the node, its count, its determiner and, if present, search for the most similar object in the current context and merge it with the one currently being built. If there's no determiner, just create a new object and push it to the context.

5) Parser context: The parser context keeps track of the detected objects from the sentences, and has a relevance hierarchy of the objects. Simply put, the relevance hierarchy is a stack of most recently mentioned objects in the current sentence. It's used for finding the most relevant item in evaluation of the parse tree. The key idea is that when speaking in natural language, we don't explicitly restate everything we said before, just the most relevant details. For instance, in the sentence *"There is a big black table next to a fridge, and there is a vase on top of the table"*, we don't need to say the vase is on the big, black table, because we inherently understand which table we are talking about. Other times, we just use pronouns, like in the sentence *"There is a table and a vase on top of it"*. As we search down the context hierarchy, we increase the penalty for matching those objects, because as we go further down, the probability we were thinking of those objects decreases.

6) Dealing with non-supported sentences: Every system has to be robust enough to deal with non-supported inputs, and so has this one. The system must handle non-supported sentences, be it wrongly-entered ones, or simply not supported by the grammar. We solve this problem with a database of correctly entered sentences.

Every supported sentence is stored in the database after being successfully parsed and tokenized, as a tokenized sentence. If a non-supported sentence is entered, the sentence database is used to generate a highest-matching sentence to the entered one. The criteria for closest sentences is the minimum edit distance between their sentence tokenizations.

For efficiency, the database is organized as a prefix tree (trie) [8], where each edge corresponds to a token, and a path corresponds to a contiguous subsequence of a tokenized sentence.

C. Model finder

After the sentenced are parsed and the object data is extracted, a corresponding 3D model for each object, as well as for the scene is created by the model finder. The model parser parses all models from the model database, which is Stanford's SceneSynth [4] dataset. In addition, it also calculates useful statistics from some of the pre-generated scenes in the dataset, which are used to deduce the dimensions of the models, and some implicit relations.

1) SceneSynth model dataset: The dataset we have used for this system is Stanford's SceneSynth model dataset. It contains 1741 distinct 3D models, 18 of which are rooms. All of the models are annotated with a name and several tags, which describe them. The dataset also includes 133 pre-made scenes, which we use for statistical learning of the parameter *size* for the models and some implicit relations.

2) Matching objects with models: Finding the most suitable model for the object is straightforward: we calculate the similarity between the object's attributes and the model's tags for each model, and pick the one which corresponds to the given object the most. The similarity function for lists of strings is the following:

$$\begin{aligned} S(x,y) = & \frac{\sum_{w\in x} \max_{v\in y} sim(w,v)}{2} \\ & + \frac{\sum_{w\in y} \max_{v\in x} sim(w,v)}{2} \end{aligned} \tag{1}$$

The sim function is the Wu-Palmer similarity measure for two strings [9], which is available in the NLTK library.

IV. Conclusion and future work

In this paper, the steps of building the NLP part of our text-to-scene genrator were presented. One can expect that a more sophisticated text analysis could perform better, although sometimes simple and shallow processing gives satisfying results, if not better [10].

The initial system we propose is just an exploratory study and further improvements are envisioned. In particular, the grammar could be expanded to support more sentences. The sentence parser could be extended with an AI-based relationship extractor, which could help in parsing sentences, which ca not be recognized by the grammar parser. Creating large annotated dataset from previous successful cases generated by the text-to-scene generator could improve the performance, with a potential use in projects with similar goals. In this paper, we have highlighted only two components of the system, namely, the sentence parser and the model finder. The scene visualizer is yet to be done. One idea is to create a Blender script that will draw objects in Blender [11], and provide a Blender file as an output.

V. Acknowledgements

This work was partially financed by the Faculty of Computer Science and Engineering at the Ss. Cyril and Methodius University, Skopje, North Macedonia.

References

[1] B. Coyne and R. Sproat, "Wordseye: an automatic text-to-scene conversion system," in *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*. ACM, 2001, pp. 487–496.

[2] A. X. Chang, M. Eric, M. Savva, and C. D. Manning, "Sceneseer: 3d scene design with natural language," *arXiv preprint arXiv:1703.00050*, 2017.

[3] A. Chang, M. Savva, and C. Manning, "Semantic parsing for text to 3d scene generation," in *Proceedings of the ACL 2014 Workshop on Semantic Parsing*, 2014, pp. 17–21.

[4] M. Fisher, D. Ritchie, M. Savva, T. Funkhouser, and P. Hanrahan, "Example-based synthesis of 3d object arrangements," in *ACM SIGGRAPH Asia 2012 papers*, ser. SIGGRAPH Asia '12, 2012.

[5] "Matplotlib: Python plotting." [Online]. Available: https://matplotlib.org/

[6] "spaCy - Industrial-strength Natural Language Processing in Python." [Online]. Available: https://spacy.io/

[7] "NLTK Book, Chapter 8: Analyzing Sentence Structure." [Online]. Available: https://www.nltk.org/book/ch08.html

[8] "Trie," Jan 2019. [Online]. Available: https://en.wikipedia.org/wiki/Trie

[9] "Words similarity/relatedness using WuPalmer Algorithm." [Online]. Available: https://blog.thedigitalgroup.com/words-similarityrelatedness-using-wupalmer-algorithm

[10] P. Gamallo, "An Overview of Open Information Extraction (Invited talk)," in *3rd Symposium on Languages, Applications and Technologies*, ser. OpenAccess Series in Informatics (OASIcs), M. J. V. Pereira, J. P. Leal, and A. Simões, Eds., vol. 38. Dagstuhl, Germany: Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2014, pp. 13–16. [Online]. Available: http://drops.dagstuhl.de/opus/volltexte/2014/4555

[11] B. Foundation, "Home of the blender project - free and open 3d creation software." [Online]. Available: https://www.blender.org/

[12] "Princeton modelnet." [Online]. Available: http://modelnet.cs.princeton.edu/

[13] S. R. Team. [Online]. Available: https://www.shapenet.org/

[14] "Natural language toolkit." [Online]. Available: https://www.nltk.org/

[15] A. Chang, M. Savva, and C. D. Manning, "Learning spatial knowledge for text to 3d scene generation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2014, pp. 2028–2038.

Last Mile Delivery with Autonomous Vehicles: Fiction or Reality?

Sasho Gramatikov
Faculty of Comp. Sci & Eng.
Ss. Cyril and Methodius University
Skopje, North Macedonia
sasho.gramatikov@finki.ukim.mk

Ivan Kitanovski
Faculty of Comp. Sci & Eng.
Ss. Cyril and Methodius University
Skopje, North Macedonia
ivan.kitanovski@finki.ukim.mk

Igor Mishkovski
Faculty of Comp. Sci & Eng.
Ss. Cyril and Methodius University
Skopje, North Macedonia
igor.mishkovski@finki.ukim.mk

Milos Jovanovik
Faculty of Comp. Sci & Eng.
Ss. Cyril and Methodius University
Skopje, North Macedonia
milos.jovanovik@finki.ukim.mk

Abstract—**Autonomous vehicles (AVs) are a disruptive technology of the 21-st century that are beginning the next revolution of the transportation of people and goods. Their presence has a particular impact on the future directions of development of E-commerce. The number of online orders is in a steep incline, and so is the necessity to deliver goods to the customer in an efficient and environmental friendly way. Using autonomous drones, pods and vans for delivery of goods has already become reality [1]. But, what is the state of the art of the companies offering these services and how do people feel about it? The aim of this paper is to make an overview of the business models of the companies developing AVs for Last Mile Delivery (LMD) of goods and to find out what is the attitudes of the online customers towards using AVs for delivery of their goods.**

Index Terms—**Autonomous vehicles, Last mile delivery, Business Model, Survey, Public opinion**

I. Introduction

With the growth of the E-commerce as a common global practice for shopping, the number of on-line orders has considerably increased, as have the expectations of the customers for faster last mile delivery. In the pursue of solution for satisfaction of these expectation and in the struggle for greater market share, the leading companies have dedicated a considerate effort and money to find more innovative and efficient ways of last mile delivery of packages [2]. One such solution is delivery of packets to customers using Autonomous Vehicles (AVs) [3]. These vehicles have become a reality due to the advances of electric vehicles technology, computer vision and machine learning.

We can already witness drones in the skies delivering packets above the crowded cities [4] or autonomous pods wandering around the pavements of the cities carrying worm meals from the restaurants. Although many such vehicles are in a pilot phase for becoming approved and secure delivery solution, there are already companies that launched AVs that offer commercial delivery services. Since the LMD with AVs is in pilot phase only in few developed countries, one of the goals of this papers is to identify these companies and give a general picture of their business model.

Although these ambitious ideas for LMD are not far from becoming common practice, there is still public skepticism for their full incorporation regarding the safety concerns imposed by letting driver-less drones, pods or vans make decision on their own in busy roads and crossroads or crowded pedestrian areas. Other concerns that affect customers are the security of the goods delivered, the privacy of the customers. In order to sense the public opinion for acceptance of AV in LMD, we conducted a survey that compares the e-commerce customers preference for delivery of their orders via AVs as opposed to traditional delivery.

The rest of the paper is organized as follows. In section two we present an overview of the business models which companies use for AVs in LWD. The survey to gauge the public opinion on AVs in LWD is presented in section three, along with the results. A discussion about the result is laid out in section four. Finally, we present the concluding results in section five.

II. Last Mile Delivery Business models Evaluation

In order to evaluate the current state of AVs for LMD, we searched for companies that have autonomous delivery of goods as their primary revenue model. As a source, we used the publicly available data on the Internet, such as the official web sites of the companies, news articles and blogs. The companies included in the analysis were classified based on whether they use drones (Figure 1) or ground based vehicles such as pods and vans (Figure 2). We considered Amazon Prime Air Delivery [5], Wing [6], Tactical Robotics [7] and Drone Delivery Canada [8] in the first group, and Starship Technologies [9], Marble [10], Robby Technologies [11], Nuro-R1 [12], Dispatch Carry [13] and EZ-Pro [14] in the second group. The aggregated results for the properties of the business model canvas are shown in Table I.

Fig. 1. Drone-based delivery: Amazon Prime Air

Fig. 2. Ground-based delivery: Robby Technologies

TABLE I
OVERVIEW OF BUSINESS MODEL PROPERTIES OF LMD COMPANIES

	Ground-based delivery	**Drone-based delivery**
Year founded	2014-2018	2012-2018
Type of company	Start up	Existing company
Country	USA, England, Germany, France	USA, Canada, Israel
Radius	up to 10 km	up to 150 km
Freight max. weight	/	1.5 - 500kg
Type of packages	Groceries, food, packets	Packets, cargo
Security	Yes	Yes
Fully implemented	Yes	Yes
Assistance	Mostly no	No
Model	B2B	B2B and B2C
Revenue model	Retailer paid	Consumer and retailer paid
Funding	$55M-$1B	/
Cost	/	/

According to the table results, the ground-based delivery business has be founded in the past few years by start-up companies with considerate investments. Their business model is B2B, i.e., they offer services to retailers that need delivery of goods to their costumers, mostly groceries or meals from local restaurants. Their vehicles cover ranges of up to 10km and, with a few exceptions, deliver the goods with no assistance. The recipient is usually notified for the arrival of the vehicle, and upon insertion of a security pin, the packet can be withdrawn from the vehicle. In the case of an assisted vehicle, the human operator only delivers the package from the vehicle to the door of the customers. The companies charge the retailers for the delivery service. On the other hand, the drone-based delivery business is a few years ”older” and has been generally founded by wealthy and well known companies (hence the lack of funding data) with the aim to cover larger regions of urban areas and to deliver heavier packets, which, in some cases, can be classified as cargo. In their business model, besides offering service to the retailers, they also offer delivery service directly to the customers. Therefore, the companies can also directly charge the customer for delivery of the goods. Both groups provide mobile applications, so the the customers can get notified and track where their delivery are in real-time. The vehicles are completely autonomous, but constantly monitored by ground operators who can take the control in critical situations. All the companies are founded in developed countries and they all have full implementation in the current year or expect to go commercially next year.

III. SURVEY

In order to evaluate the preference of people to use AV delivery as opposed to the traditional delivery, we conducted a short online survey [15], [16]. The survey consisted of different groups of questions which aim to profile the customers according to their socio-demographic characteristics and their attitudes toward AV last mile delivery. The structure of the survey is depicted on Figure 3.

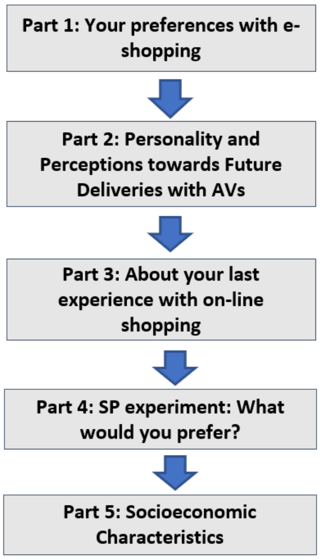

Part 1: Your preferences with e-shopping
Part 2: Personality and Perceptions towards Future Deliveries with AVs
Part 3: About your last experience with on-line shopping
Part 4: SP experiment: What would you prefer?
Part 5: Socioeconomic Characteristics

Fig. 3. Survey structure

The survey was conducted in March 2019 within a period of 2 days. The majority of the participants were attendants of a Training school related to the autonomous connected transport. The questionnaire was completed by 33 participants form 12 countries with average age of 38.1 year. Most of the participants had first or second degree of education. The participants were gender-balanced.

The first group of questions profiles the participants as online-shoppers. Figure 4 shows the frequency of using delivery services when shopping on-line different types of products. From the figure we can see that when purchasing big size products or food from the supermarket the participants seldom use delivery, or use it only a few times a year. The frequency is slightly higher for delivery of small sized products (books, electronics, toys, etc), clothes and shoes, which ranges between never and once a week. What is most frequently delivered to the users is food from a restaurant. The results show that the customers still do not receive their everyday shopping items directly in their homes or offices.

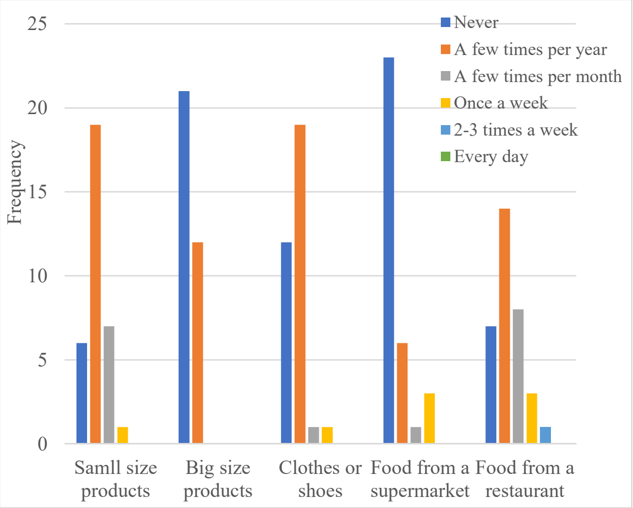


Fig. 4. Frequency of using delivery services for online buying of different types of products

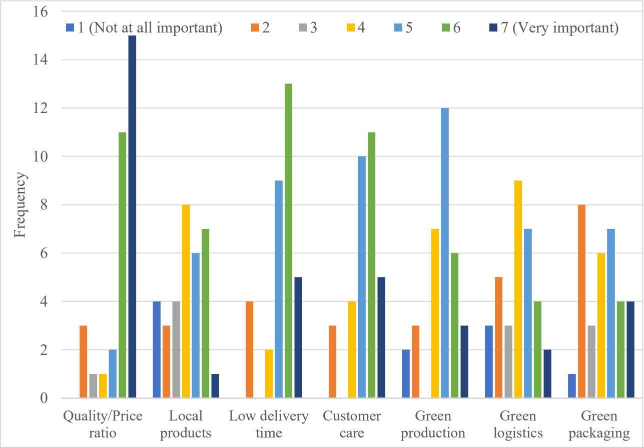


Fig. 5. Perception of importance when buying a product online

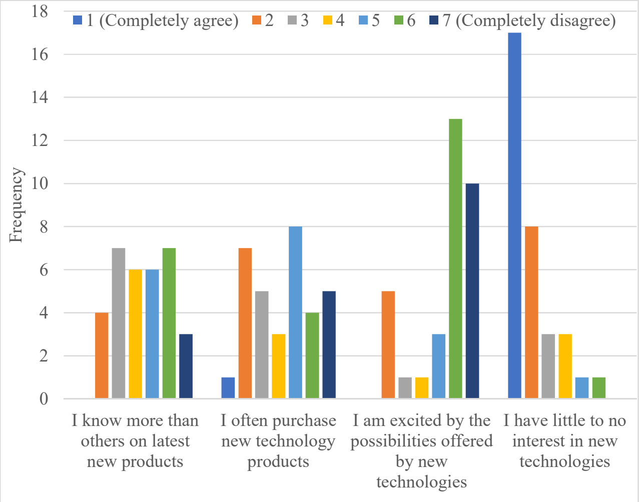


Fig. 6. Customers' attitude towards new technologies and products

Figure 5 shows the perception of importance when buying a product on-line. The most important aspect of on-line shopping for the participants is the quality/price ratio of the product, the delivery time and the customer care, which means that they want to get cheap products with good quality in a very short time, and at the same time, to be sure that they can get the best of what they payed for. The participants have shared opinion regarding the locality of the products: some of them prefer that the products are local so that they can be delivered faster, while some give no importance to the origin of the product, since what matters *most* is its price. Most of the participants also give *some* importance to the environmental impact of e-commerce, from the perspective of production, packaging and logistics.

Since delivery with AVs by itself is a new technology, the participants were asked questions that reveal their attitudes towards new products and technologies. From the results presented in Figure 6, we can conclude that the most of the participants have big interest in new technology and are excited by the possibilities they offer. However, they have different levels of knowledge related to the new technologies.

When asked about their perceptions towards future deliveries with AV, as Figure 7 shows, the majority of participants would greatly value fast delivery with AVs, although there are some that do not give much credit to the whole idea. Nevertheless, they mostly disagree that the delivery with AVs will cause them to lose control over their personal security. The fact that the AVs are unattended when moving in the open space leaves many doubts about the safety of the packets on their way to the customers, which can be seen by the responses of half of the participants. On the other hand, as shown in the analysis of the business models of the LMD companies, the packets are secured with a unique code shared with each customer separately, so that only the customer, can open the case upon arrival. Therefore, another half of the participants do not fear that the packet may be stolen. Similar distribution of responses can be observed on their opinion about the data privacy with AVs deliveries. Despite of these concerns, a vast majority of the online customers would use AVs to receive their orders in the future.

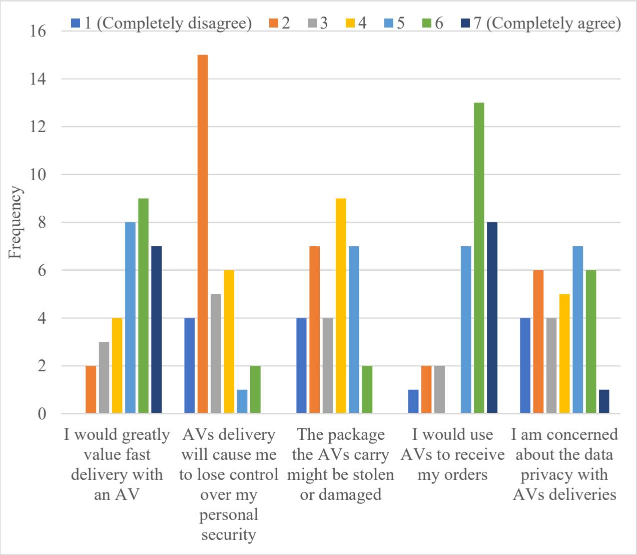


Fig. 7. Perceptions towards Future Deliveries with AV

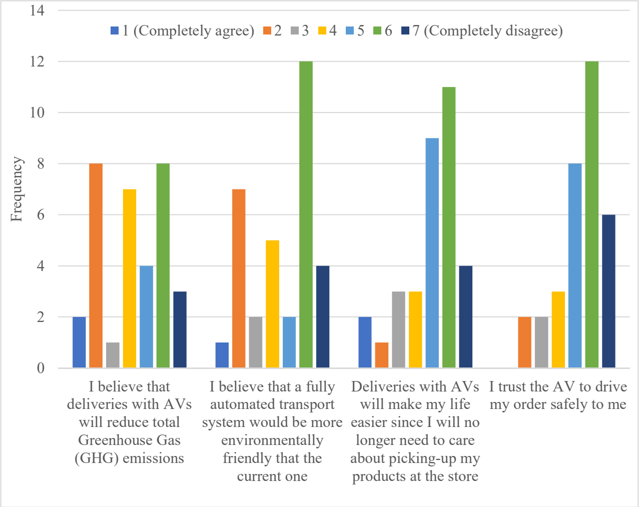


Fig. 8. Perceptions towards Future Deliveries with AV and the environment

The positive attitude of the participants towards the AVs is also confirmed in the results shown in Figure 8. The AVs are highly trusted to safely deliver the orders to the customers. Moreover, they believe that this mode of delivery will considerably make their life easier when shopping is concerned, since they do not have to leave the commodity of their homes in order to pick-up their products.

In the survey, the participants were also asked about the impact of the AV delivery on the environment. In Figure 8, we can see that they mostly believe that the automated transport system would be more environmentally friendly than the traditional delivery system. However, not all the participants believe in such a scenario. The situation is also quite similar in the responses regarding their believes that the AVs will reduce the greenhouse gas emissions. One reason for the skepticism, despite the fact that the AVs are electrically powered, is that they do not see that the growth of the autonomous vehicles will come to a scale that will dominate over the traditional transport, so that AV-based delivery can have any global effect.

IV. Discussion

The results show that the participants generally keep up with the new technologies and are eager to use the latest tech-products. However, for certain products, they still do not use delivery services for on-line purchase as often as it would be expected for a future where most of the purchases will be made on-line. As the participants have never had experience with AV delivery, a certain part of them are unsure about using AVs in their future deliveries, fearing that their packet will be stolen, or that they would have issues with their privacy. The lack of AVs delivery may be a good reason for such attitude, since the customers are not acquainted with the efforts that the companies make to provide the best of the delivery service. One should also note that the average age of the participants is rather high, and hence, different results may be expected from younger generations which have grown with high technologies from their earliest ages. It should also be mentioned that the survey was conducted for only a short period of time and distributed only to a small group of people and their friends with same level of education, which resulted in a relatively small number of participants which is not high enough to represent the global opinion of the general public.

V. Conclusion

The AVs are already at our doorways. The companies are intensively introducing them in the delivery services, complying with the increasing demands of innovative and efficient delivery for the E-commerce. From our on-line company search, we can conclude that the companies are keeping up with these demands by offering drone-based and ground-based delivery with AVs. They offer many features such as security, safety, speed, environment awareness and, most importantly convenience, which make them a promising substitution to the tradition transportation for delivery. The same conclusion can be obtained from the survey that we conducted on a small group of e-commerce customers. Most of the participants of our survey accept and strongly believe in the future of the AVs. They think that AVs will make their life easier without fearing their own safety or the safety of the goods they order. They also find the AV delivery more environmental friendly than the tradition one. However, there is still a minority that feels doubtful about the AVs and their benefit. Since the most important aspects for all the participants are good quality/price products, fast delivery and customer care, and at the same time, there are participants that are not convinced about the safety of the products and their privacy, the current and the future companies dealing with the AV delivery services will have to consider offering acceptable prices for a fast service, guaranteeing the safety of the goods, the costumers' data and all the participants in the traffic. At this day, the AVs are still under doubts by many people, just like the traditional vehicles were less than a century ago. With this in mind, if we go froward to the future, we may be quite sure that AVs will conquer the market of global transportation, leading the modern civilization to next level of progress. .

VI. Acknowledgment

The authors thank the Faculty of computer science and engineering at the Ss. Cyril and Methodius University in Skopje, under the Flow2OD (" Generation of universal simulation data based on real traffic flow data ") project for financial support.

References

[1] M. Slabinac *et al.*, "Innovative solutions for a last-mile delivery–a european experience," in *Proceedings of the 15th International Scientific Conference Business Logistics in Modern Management Osijek, Osijek, Croatia*, 2015, pp. 111–129.

[2] H. L. Lee, Y. Chen, B. Gillai, and S. Rammohan, "Technological disruption and innovation in last-mile delivery," *Value Chain Innovation Initiative*, 2016.

[3] M. Joerss, J. Schröder, F. Neuhaus, C. Klink, and F. Mann, "Parcel delivery: The future of last mile," *McKinsey & Company*, 2016.

[4] W. Yoo, E. Yu, and J. Jung, "Drone delivery: Factors affecting the publics attitude and intention to adopt," *Telematics and Informatics*, vol. 35, no. 6, pp. 1687–1700, 2018.

[5] (2019) Amazon Prime Air Delivery. [Online]. Available: https://www.amazon.com/Amazon-Prime-Air/b?ie=UTF8node=8037720011

[6] (2019) Wing project. [Online]. Available: https://x.company/projects/wing/

[7] (2019) Tactical Robotics Cormorant. [Online]. Available: http://www.tactical-robotics.com/category/cormorant

[8] (2019) Drone Delivery Canada. [Online]. Available: https://dronedeliverycanada.com

[9] (2019) Starship Technologies. [Online]. Available: https://www.starship.xyz/

[10] (2019) Marble. [Online]. Available: https://www.marble.io

[11] (2019) Robby Technologies. [Online]. Available: https://robby.io

[12] (2019) Nuro-R1. [Online]. Available: https://nuro.ai

[13] (2019) Dispatch Carry. [Online]. Available: https://dispatch.ai

[14] (2019) EZ-pro. [Online]. Available: https://nuro.ai

[15] A. Polydoropoulou, I. Pagoni, and A. Tsirimpa, "Ready for mobility as a service? insights from stakeholders and end-users," *Travel Behaviour and Society*, 2018.

[16] M. Kamargianni, M. Ben-Akiva, and A. Polydoropoulou, "Incorporating social interaction into hybrid choice models," *Transportation*, vol. 41, no. 6, pp. 1263–1285, 2014.

Matching Economic and Enviromental Aspects of Energy Efficiency in Single ICT model

Igor Bimbiloski
Faculty of Electrical Engineering and Information Technology
Univ. Ss. Cyril and Methodius, Skopje, Macedonia
igor.bimbiloski@gmail.com

Aleksandar Risteski
Faculty of Electrical Engineering and Information Technology
Univ. Ss. Cyril and Methodius, Skopje, Macedonia
acerist@feit.ukim.edu.mk

***Abstract* — One of the reasons for not having enough investments in green energy systems is the financial rationale behind. ICT industry is technologically ready to satisfy the requirements of energy industry and policy makers, but finding the appropriate model that will match the economic and environmental interest is a challenge, as well as its implementation. This paper is presenting a model that could satisfy the opposed economic and environmental interest, and match them in the single ecosystem in the single time. The model is using existing ICT technology resources, and game theory analysis. The analysis shows that the model is financially viable for all players on the energy market. Also, it's introducing possibility of transfer of financial into environmental benefits, and supporting the investment in green energy systems.**

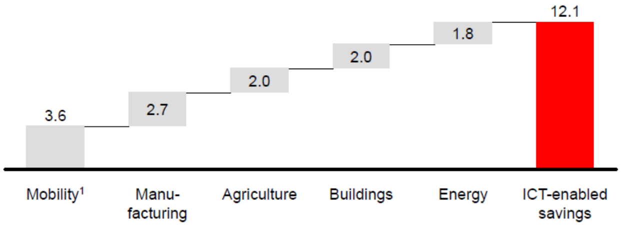
Keywords — Economic vs Environmental Goals; Energy Efficiency; ICT model; Game Theory; Competitive and Cooperative approach.

I. INTRODUCTION

Among the 17 sustainable development goals defined in the 2030 program of the United Nations (UN), the fight against climate change is treated as one of the biggest challenges of our time, whose negative consequences make it impossible for the economy to achieve sustainable development. The fight against climate change from this aspect is not an option but necessity. Without reducing greenhouse gas emissions and preventing global warming, many other global problems such as poverty, hunger, water scarcity, etc. will become much more difficult to resolve. At the moment the planet faces serious global warming, which is most likely (over 95%) a consequence of human activity [1]. In the mid-20th century, it reached its historical maximum and continues to grow at a worrying pace in the future of the planet.

The solution to this problem is developed in two directions: the first is to mitigate climate change by reducing emissions and stabilizing levels of greenhouse gas pollution in the heat-retaining atmosphere, and above all on CO_2, and the second direction of adaptation of humanity to the new planetary conditions, primarily using scientific and technological means. One of the most commonly mentioned solutions is finding innovations in the field of energy, as for new sources of energy, energy storage, renewable sources, smart energy networks and energy efficiency systems [2].

According to the Global e-Sustainability Initiative (GeSI), ICT has the potential to reduce greenhouse gas emissions by 20% by 2030, by helping companies and consumers for more intelligent use and energy saving. Louis Neves, chairman of GeSI, is optimistic about the ability of the industry to be fully viable. He says: "Our findings show that with ICT support, the world by 2030 will be cleaner, healthier and more prosperous, offering greater opportunities for individuals everywhere." Louis Neves says that emissions that are avoided using ICTs are already ten times greater than the emissions generated by their installation [3] As the graph below shows (Fig. 1), the sector can help to avoid the production of about 12 Giga tones CO_2 by 2030, and to stop further emission.



1 Mobility solutions consider ICT-enabled improvements to private and commercial mobility and additionally consider the reduced need to travel from various sectors, including health, learning, commerce, etc.
Source: WRI, IPCC, World Bank, GeSI, Accenture analysis & CO2 models

Fig. 1. CO_2 abatement potential by sector

Since all three major economic sectors, that is, industry, transport, and buildings have high energy use, energy conservation is a critical task. Buildings consume a significant share of global energy consumption. Therefore, significant energy savings can be realized from facilities that are properly designed and managed. The rapid increase in energy consumption in recent decades has also resulted in an increase in living standards. Namely, the facilities now spend more than 30% of the total energy in the world [4]. In Europe, the building sector accounts for 40% of energy consumption and 36% of emissions of CO_2 [5], and in the United States, the building sector with 38.9% of total primary energy demand [6]. In addition, for a modern city like Hong Kong, 60% of carbon emissions are due to the generation of electricity, of which 89% of the total electricity consumption for the needs of buildings [7]. Hence, increasing the energy efficiency and energy performance of buildings is essential to alleviate the increased demand for additional power supply as well as the emission of CO_2.

Our goal is to find a model that will satisfy and match the current environmental goals of the ICT industry with the investments goals in the green energy. The model is using available technologies and game theory approach as a mathematical tool to prove the coexistence of both aspects in single environment.

II. DIVERSITY OF GOALS IN SINGLE SYSTEM

As we have already mentioned above, the complexity of the issue imposes on defining several goals, with a different nature: technologic, economic and social. Above all, the following goals are set:

- What model to use, from which elements it consists, and how are they interconnected, what are the rules of physical and logical connection, and under what rules does the system work?

- What methodology will be used to verify, from a scientific point of view, the viability and applicability of this solution?
- The proposed solution should be technologically and economically viable, that is, acceptable to all stakeholders, easy to implement and use, and be market-impulsive?

If we take a closer look at the situation of ICTs and the energy industry, we can define the following goals by segments:

1) Technological goal - to find a solution that is technologically feasible and which can use already proven and validated technologies. Thus, a model uses several technological segments:

a) Telecommunications segment, or network segment, in which reliable technologies should be found that will provide reliable data transfer. It is proposed to use two types of networks, namely traditional networks of already existing operators, such as fourth generation networks (4G networks), and networks that will have additional options for trusted and IoT data transfer, which are 5G networks.

b) Information systems that can be used to smartly manage the energy infrastructure, but above all they can coordinate the habits and activities of consumers. Artificial intelligence systems for managing user habits are already being used (for example, Facebook, Google, Alibaba ...) primarily for marketing purposes, so the same technology is proposed to be used in this energy-efficient model that is user-oriented.

c) User terminal equipment, which should be affordable and easy to use. In our model, we recommend the use of smartphones and mobile applications, which implies that the final consumer does not need additional investment, but just plain installation of a free mobile application, which will be easy and simple to use at the same time.

2) Economic goal - to find a solution which in principle should satisfy the following characteristics:

a) Do not require large investments and be easy to implement

b) To have a reasonably high degree of cost-effectiveness, i.e. to have a quick return on investment

c) be propulsive on the market, that is, it can be quickly expanded to many consumers

3) Ecological dimension i.e. social sense for the implementation of the solution, which aims at:

a) Reduction of greenhouse gas emissions and changes in climatic conditions, i.e. environmental protection through the use of the solution and proposed model.

b) Change of consumer awareness, behavior and habits, which would lead to a level of self-sustaining ecological stability.

To meet all the goals of the proposed solution, the model uses the theory of games as a universal approach and a methodology for proving these goals by the proposed model. Namely, the theory of games is the methodology for strategic positioning of the entities in a given environment, and analysis of the possible scenarios and the benefits or losses in those scenarios. It has a wide range of uses, from military, so economic, social and most recent technological goals. Hence, taking into account the diversity of the goals we defined earlier, the theory of game will be used as a scientific tool for confirming the correctness of the proposed ICT model for energy efficiency.

III. Proposed Model

In the proposed model MITS , the end user i.e. the consumer is in the center of the energy ecosystem and the proposed model is consumer-oriented model. Changing the user behavior versus big energy efficiency investments is increasingly in the focus of energy policy makers and regulatory bodies, which is an indication that the proposed model could receive institutional support. In October 2012, the European Commission adopted a Directive to reduce energy consumption by 20% by 2020 [8], which identified the potential for energy savings by changing the behavior of user habits, which could result in savings of 5 up to 20%.The assumption is that the consumer can be in three types of state (state of movement, privacy and at work [9-10], according to which the appropriate energy efficiency model is determined. The idea also includes the use of smartphones as a tool to identify the state of the consumer, while at the same time it will also serve as a communication tool through which directions and advices (messages, information, etc.) will be received about the behavior of the consumer. Considering the everyday use of smartphones and mobile applications, as well as user dependency and time of use, it is expected that their use will strongly affect energy efficiency habits.

The draft model will use the two common basic elements: consumer and supplier. The first element - the consumer i.e. man will be identified through the smartphone and it can be located in three states: state of work, a state of privacy and a state of motion, as shown in Fig. 2. The second element is the supplier, who will be identified through the facility and the electric meter in business or private facilities or vehicles for transporting energy consumers. Accordingly, we will have three forms of suppliers of metering energy: a supplier of business facilities, a supplier of private facilities and a supplier of consumer transport. As it was explained in the previous chapter (Fig.1) these activities are considering almost 60% of the target industries.

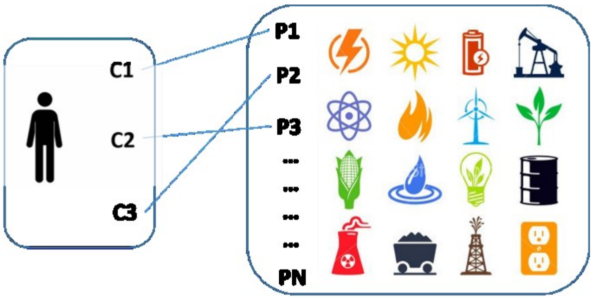


Fig. 2. Connection of the consumer state with the energy supplier

The consumer C can be located in one of the three states that can change over time t. For example, when the C_1 consumer is in a state of work, he uses the energy of the supplier P_1. The provider cannot change the state, but he has a predefined state. On the supplier side P_1, there is a measuring device that measures the energy consumption, where

$$\mathrm{P}_{1(C_1)} = f(\{C_1\}, t)$$
$$\mathrm{P}_{2(C_2)} = f(\{C_2\}, t)$$
$$\mathrm{P}_{3(C_3)} = f(\{C_3\}, t) \tag{1}$$

Since in the time t_0 there may be *N* consumers connected to the same meter, the total energy spent on the P_1 provider will be:

$$\mathrm{P}_1 = \sum_{x=1}^{N} \mathrm{P}_{1(C_x)} \tag{2}$$

And the total energy consumption among all *M* providers will be

$$\mathrm{P}_{total} = \sum_{x=1}^{M} \mathrm{P}_x \tag{3}$$

All parameters on the supplier's side $\mathrm{P}_1\ to\ \mathrm{P}_M$ are measurable in time. On the left side i.e. consumption side, only the cell in which the consumer is located is known, and it is true that the energy used is equal to P_{total}.

$$\sum_{\substack{x=1\\y=1}}^{\substack{N\\M}} \mathrm{P}_{\mathrm{x}(C_y)} = \mathrm{P}_{total} \tag{4}$$

The purpose of the model will be to make a prediction for:

- for the consumer's state according to the time

$$\mathrm{C}_{\mathrm{x}(t_a)} = C_1\ or\ C_2\ or\ C_3 \tag{5}$$

- the location on which the object is located – the supplier from which it uses energy

$$\mathrm{P}_{\mathrm{x}} = f(C_1, geolocation) \tag{6}$$

- the amount of energy it uses

$$\mathrm{P}_{\mathrm{x}(C_y)} = f\{C_{y,}\ time\} \tag{7}$$

IV. Game Theory Analysis

If we use the Game Theory Analysis in the competitive environment with multi agent smart market [11], we will gain positive payoffs on both consumer and provider side.

We proved [11] that the general conditions for positive payoff of the provider are:

$$M_{cfn}(j, t_r) = \sum_{c=1}^{C}\sum_{j=1}^{J}\sum_{r=1}^{R}\left\{\begin{array}{c} \left\{\begin{array}{c} [p_{nr} - i_{fn}(j,t_r)] * l_c(j,t_r),\ when\ i_f(j,t_r) < i_c \\ [p_{nr} - i_c] * l_c(j,t_r),\ when\ i_f(j,t_r) > i_c \end{array}\right\} \\ when\ p_{nr} < p_{mr}\ for\ m \neq n\ and\ m = \{1,2,..,N\} \\ \\ 0\ \ \dots\dots\dots\dots\dots..\ \ in\ any\ other\ case \end{array}\right\} \tag{8}$$

where $M_{cfn}(j, t_r)$ is payoff of the provider in J days with t_r timeslots for price change, $= \{1, \dots, F\}$ is the number of the providers, $c = \{1, \dots, C\}$ is the number of consumers, $(p_f(j,t))_{j\in \mathrm{J}, t\in \mathrm{T}}$ is the price towards consumers, $i_f(j,t)$ is the cost price of electricity, $l_c(j,t_r)$ is the volume of electricity, and *m,n* ∈ *f*, i_c is the agreed price for buying green electricity from consumers.

Consequently [11], for the consumer the conditions for positive payoff of MITS usage is:

$$\Delta B_{cf}(j,t_r) = \sum_{\mathrm{j}=1}^{\mathrm{J}}\sum_{r=1}^{R}\sum_{n=1}^{N}\left\{\begin{array}{c} p_{nr} * \Delta\, l_c(j,t_r); \\ when\ p_{nr} < p_{mr}\ for\ any\ m \neq n\ i\ m = \{1,2,..,N\} \\ 0\ \ \dots\dots\dots\dots\dots..\ \ any\ other\ case \end{array}\right\} + \sum_{w=1}^{R}\sum_{q=1}^{R}\sum_{m=1}^{N}\sum_{n=1}^{N}\left\{\left\{\begin{array}{c} (p_{wn} - p_{qm}) * \Delta\, l_c(j,t_{w\to q})\ \text{when}\ p_{wn} > p_{qm} \\ 0,\ \text{when}\ p_{wn} \le p_{qm} \end{array}\right\}\right\}$$
$$+ \sum_{\mathrm{j}=1}^{\mathrm{J}}\sum_{r=1}^{R}\sum_{n=1}^{N}\left\{\begin{array}{c} p_{nr} * \Delta\, l_c(j,t_r); \\ when\ p_{nr} < p_{mr}\ for\ any\ m \neq n\ i\ m = \{1,2,..,N\} \\ 0\ \ \dots\dots\dots\dots\dots..\ \ any\ other\ case \end{array}\right\} -$$
$$I_{p,s} + \sum_{j=1}^{J}\sum_{r=1}^{R}\sum_{n=1}^{N}\left\{\begin{array}{l} i_c * l_{sale}(j,t_r), \\ when \left\{\begin{array}{c} i_{fn}(j,t_r) > i_c \\ \text{и} \\ l_{prod.}(j,t_r) > l_{sale}(j,t_r) \end{array}\right\} and\ \ i_{fn}(j,t_r) > i_{fm}(j,t_r)\ \ for\ m \neq \\ n,\ m = \{1,2,\dots,N\} \end{array}\right\} \text{-}\ I_{sale} \tag{9}$$

where $t_{w\to q}$ is transfer of consumption from w to q timeslot, $I_{p,s}$ is the investment in facilities for production and storage of electricity on home level, for internal home use, I_{sale} is the value of the investments for the distributed network electricity provisioning.

Matching it with cooperative game theory [12], we proved that the cooperative approach gave stable core as a solution for the game and that the payoffs from the competitive game are part of the core of cooperative game (Fig.3). That proves that the model is a single stable model with combined effects: competitive (to bring financial benefits) and cooperative (to support environment protection).

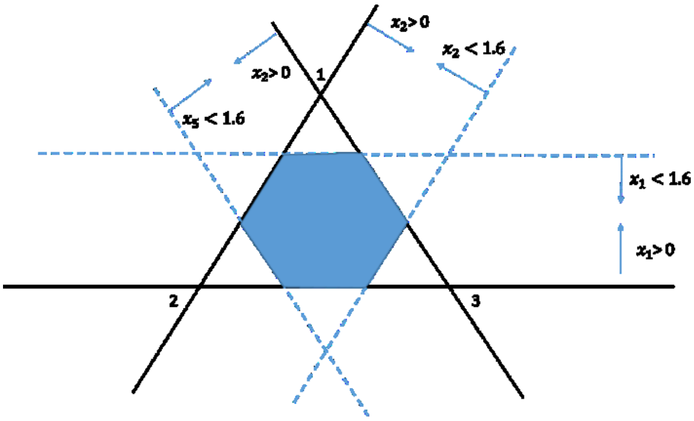


Fig 3: Core plane of the model

What is also important, the possibility to use the Shapley rules of the fair distribution of transferable benefits, what we are presenting in this paper. Actually, if we have a large number of consumers *M*, and more operators *N*, where *M* >> *N* (the number of consumers in the markets is measured in millions and the number of providers is limited), then the question is how to balance the benefits according to principles of Shapley. If we have a cooperative game with *M* + *N* players, where S is a coalition of players in the game $S \subset (N+M)$, and if the assumptions we have considered before apply to it, then for Shapley's formula we get that

$$\Phi_i(\mathrm{M}+N,v) = \sum_{S \subseteq N\{i\}} \frac{|S|!\,(|\mathrm{M}+N|-|S|-1)!}{|\mathrm{M}+N|!}\ [v(S \cup \{i\}) - v(S)] \quad (10)$$

where $[v(S \cup \{i\}) - v(S)]$ is the marginal contribution of the player $\boldsymbol{i}$ in the coalition S. If M is large enough $(M \rightarrow \infty)$, but still has a fixed value, then the allocation from the benefit to each consumer will individually weigh

$$\Phi_C(\mathrm{M},v) = (\,l(t_r) + \ g(t_r)) \qquad (11)$$

And for the providers

$$\Phi_P(N,v) \rightarrow (0) \qquad (12)$$

V. Transfer of Benefits

Considering all above results gain from the Game Theory analysis, it is obvious that the final goal of the model, to transfer the financial in environmental benefits is viable. (Fig 4).

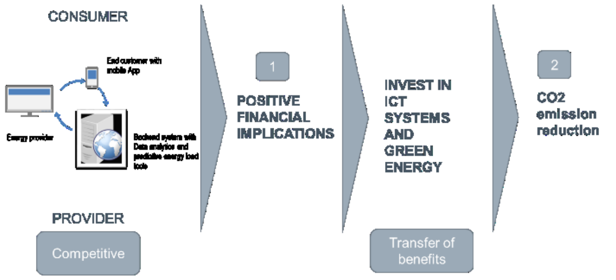

CONSUMER
End customer with mobile App
Energy provider
Backend system with Data analytics and predictive energy load tools
PROVIDER
Competitive
1
POSITIVE FINANCIAL IMPLICATIONS
INVEST IN ICT SYSTEMS AND GREEN ENERGY
Transfer of benefits
2
CO2 emission reduction

Fig 4: Transfer of financial in environmental benefits

Namely, two forms of game theory are applied: competitive and cooperative theory of the game.

Competitive game theory has been used to analyze the economic or financial benefits of using the proposed model for energy efficiency. Firstly, we have confirmed that the choice of a competitive game theory, with the features of Stockemberg's successor game, is the right model for this game, and that in such an environment all players have benefits from using the model. This game confirmed that model is used in various types of markets, from the simplest to the most developed, from energy markets with a simple one-tariff meter to smart grid markets, renewable sources and two-way transport of energy. We also showed that in model with multiple providers and consumers, the benefits of using are available.

In the next step, using the cooperative type of games, we showed that besides the financial effects with the use of model, environmental benefits can be obtained. This implies that if the market entities manage to find a model of cooperation, that co-operation increases the effect of energy savings, that is, stimulates the use of green energy instead of energy from fossil fuels.

VI. Summary

There are many scientific researches and models that treat the topic of energy efficiency, and usually each of them with different success achieves its goal. In order this model to be successful, we consider it necessary to satisfy some pre-requisites. These prerequisites, which are necessary for successful implementation, as well as its transformation from theoretical into practical environment, and which the model is satisfying are the following:

a) It should be acceptable to all parties that are present on the energy market. We achieved this in this model, because with the mathematical method of game theory we showed that model is acceptable for the final consumers, and for the providers of energy, and also contributes to reducing the emission of harmful gases into the atmosphere.

b) It should be financially profitable; i.e. it can be implemented with a small investment. And we have shown that it is valid because it uses the existing infrastructure and technology, and it is an award that connects the elements in an efficient manner. The investment in the new ICT elements of the model is insignificantly small, and it uses pre-existing injections such as smartphones and telecommunications networks.

c) It should be simple to implement, which is also achieved by this model because it is based on cloud and internet platform technology, and is available to all consumers globally or in smaller market segments. The MITS can quickly penetrate the market, because it predicts the use of a mobile application, and at the same time it's easy to use.

d) Implementation of the solution always starts on a smaller scale and grows larger, which means that the model should be scalable or upgradable. As we have seen, the model is scalable in two dimensions. The first dimension is the technical scalability. It is applicable in technologies that are simple or even obsolete, up to the most modern technologies, taking into account the latest communication technologies like 5G and IoT, like artificial intelligence, smart energy networks, renewable energy, remote control systems, etc. The second dimension is commercial scalability, or scalability on the provider's side and on the consumer side. Model is supporting markets with an unlimited number of consumers and an unlimited number of providers, as well as numerous segments of the market.

e) Model focuses on energy efficiency by coordinating the activities of providers and consumers, that is, the consumer is placed in the center of the model, and is a user-oriented system. Model achieves energy efficiency through changing user behavior, in contrast to technological investments that are financially intense in the energy sector.

f) Model is within the framework of the global policies and European regulatory processes, which means that it is ready to be and institutionally supported. Namely, in the official energy efficiency announcements of these bodies, support is already given to research and projects that focus on achieving the energy efficiency goals by changing consumer behavior by applying information-telecommunication technologies.

In this paper we tried to treat a specific technological-economic-social problem with the parallel application of a competitive and cooperative theory of games. Namely, in professional and scientific literature, there are many examples that treat the game theory and its application, but rarely combine models of game theory. With this paper, we have shown that there is not always a single approach that needs to be taken to solve problems, but that it is necessary to look for the best models that correspond to the actual situation and environmental conditions. However strange and contradicting is the use of these two game theory models, in the concrete example and the proposed ICT concept, this is exactly the best results for achieving the ultimate goals. As a final confirmation, the game theory model showed that it is possible to combine the competitive and cooperative game, to achieve different goals in the same environment at the same (parallel) time.

Another conclusion of the paper, which can be transformed from theoretically into practical, is the part to transfer of benefits from a game with competitive in a cooperative form. The paper confirms the idea that the model can give financial benefits to both providers and consumers. At the same time, there is a need for a targeted emission of harmful gases that can be solved by changing consumer habits and green energy investments. This paper, proposes that the financial benefits derived from the use of model, be used for investments in green energy sources. This solves the economic dilemma about the cost-effectiveness of green energy that we are facing today. Namely, with this model, we offer a solution by which instead of allocating additional funds for investments in green energy, through the cooperative model, the already existing budgets should be used, while the providers and consumers will not be at a loss.

In the part of the theory of games, the dilemma about the effects of a combination of various types of games in the same application is open, and the transfer of benefits from one form of game to another. With this paper we only touched this issue and we showed that several forms of game theory can exist at the same time and in the same environment, yet have a positive effect on the ultimate goals, and even they were contradicted in the initial perception of the problem. It remains to be continued to explore these combinations, the effects they cause, and generalize some conclusions if possible, as well as to see the applications where they can be used.

VII. References

[1] NASA Global Climate Change, Vital Signs of the Planet, https://climate.nasa.gov/evidence/

[2] https://www.climate.gov/news-features/understanding-climate/climate-change-atmospheric-carbon-dioxide

[3] Accenture Strategy, “#SMARTer2030 - ICT Solutions for 21st Century Challenges”, GeSI – Global Sustanability Report, 2015

[4] IEA, Key World Energy Statistics, 2009.

[5] European Parliament and Council, Directive 2010/31/EU ofthe European Parliament and ofthe Council of 19 May 2010 on the energy performance of buildings (recast), Official Journal of the European Union L153 (0) (2010) 13–35.

[6] A.G. Kwok, N.B. Rajkovich, Addressing climate change in comfort standards, Building and Environment 45 (1) (2010) 18–22.

[7] M.C. Leung, N.C.F. Tse, L.L. Lai, T.T. Chow, The use of occupancy space electrical power demand in building cooling load prediction, Energy and Buildings 55 (0) (2012) 151–163.

[8] EEA Technical Report No 5/2013, Achieving energy efficiency through behavior change: what does it take? – European Environment Agency

[9] Igor Bimbiloski, Aleksandar Risteski, “Draft Concept for Energy Efficiency Improvements with usage of Smart Phones and Artificial Neural Networks”, ENAR 2018, Alania, Turkey

[10] Igor Bimbiloski, Aleksandar Risteski, “Models for Energy Efficiency Improvement by Using Mobile Technologies and Internet of Things”, Journal of Electrical Engineering and Information Technologies, Vol. 3, No. 1–2, pp. 91–99, Skopje, 2018

[11] Igor Bimbiloski, Valentin Rakovic, Anis Sefidanoski, Aleksandar Risteski, “Competitive Game Theory Approach of Energy Efficiency ICT Model in Multiplayer Market”, accepted, 18th IEEE Eurocon International Conference on Smart Technologies, Novi Sad, Serbia, 2019

[12] Igor Bimbiloski, Aleksandar Risteski, “Matching Competitive and Cooperative Game Theory in single ICT Model for Energy Efficiency”, submited, BalkanCom 2019, Third International Balkan Conference on Communications and Networking, Skopje, North Macedonia, June, 2019

Multiple hypothesis testing: adjustment methods with application

Biljana Tojtovska
Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje
Email: biljana.tojtovska@finki.ukim.mk

***Abstract*—The goal of this paper is to present and compare few multiple comparison methods, which control the family wise error rate and false discovery rate in the case of multiply hypotheses testing. We apply the methods to data from ophthalmological research, compare the results and discuss the recommended procedures.**

***Index Terms*—Multiple comparison test, Family wise error rate, False discovery rate**

I. Introduction

Family wise error rate (FWER) is the probability of making at least one type I error (also known as false positive finding), when multiple hypothesis are tested on the same data. If α is the significance level for each test, then the probability that between n independent tests we will observe at least one significant results is

$$FWER = 1 - (1 - \alpha)^n \tag{1}$$

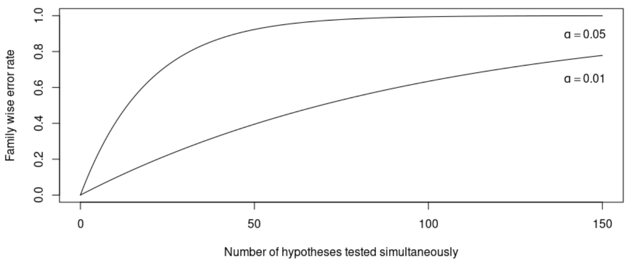


Fig. 1. The increased error rate of multiple comparisons.

As we can see in Figure 1, this probability increases fast for the standard value of $\alpha = 0.05$. For ex. for $n = 10$ there is probability of 0.4 that we will make at least one type I error, and for $n = 30$ this probability is almost 0.8. This poses a serious problem when the research asks for multiple hypothesis tests, which is standard in fields like genomics and neuroimaging. When we perform multiple ANOVA or test of independence on larger $m \times n$-contingency tables, if the test turns significant, we would need additional post-hoc tests to get more details and detect which variables lead to such a result. In many exploratory studies in a new research field it is necessary to test the relationship of larger number of variables, in order to get more details on the subject. In all these cases, with a very high probability we will obtain at least one significant result, i.e. at least one significant p-value, $p \leq \alpha$. Even though this is very desirable outcome for journal publishing, we should be aware that these may be false positives and we should proceed carefully.

Many journals ask for mandatory adjustments when multiple hypothesis are tested. In some fields there are even preferred procedures and threshold values. However, it may still be difficult to choose the right procedure, since there are a lot of things under consideration - hypothesis relationship, power of the test, effect and sample size, type of study etc. In this paper we will discuss most common multiple testing adjustments. We will divide them in two groups - Section II we present methods that control the Family wise error rate and in Section III we cover some methods that control the False discovery rate. We will describe the methods by adjusting both the threshold value α, and p-values . In Section IV we compare the methods on data from our previous study, which were published without adjustment in Ophtalmic Epidemiology [1]. Finally, we give conclusions and recommendations for future applications.

II. Procedures that control the FWER

For this and the following section, let us suppose that we test n hypothesis $H_1, H_2, ..., H_n$, each with confidence level α and let $p_1, p_2, ..., p_n$ be the corresponding p-values. We assume that between the n hypothesis, n_0 are true (see Table I) and denote $\mathcal{I}_0 = \{i | H_i \text{ is true}\}$.

TABLE I
Simultaneous hypothesis testing

	accepted	rejected	total
true	U	V	n_0
false	T	S	$n - n_0$
total	n-R	R	n

In this section we will consider procedures that control the FWER weakly or strongly. A procedure controls the FWER *weakly* if the FWER does not exceed α only when all null hypothesis are true. If the FWER is controlled independently of the testing hypothesis, then we say the procedure controls FWER *strongly*. We present here the Bonferroni, Holm-Bonferoni and Hochberg method, based on [2] and [3]. For more details on the whole subject we refer to [4], [5] .

A. Bonferroni

The Bonferroni method may be the best-known method for multiple testing [6]. With this procedure each hypothesis is tested at an adjusted significance level of α/n. Alternatively, we can define adjusted p-values by

$$p_i^{(a)} = \min\{i \cdot p_i, 1\} \quad \text{for} \quad i = \overline{1,n} \tag{2}$$

and consider significant those that satisfy $p_i^{(a)} \leq \alpha$.

For the Family wise error rate we have

$$\begin{aligned} FWER &= \mathbb{P}\Big(V \geq 1\Big) = \mathbb{P}\Big(\bigcup_{i\in\mathcal{I}_0}\{p_i < \frac{\alpha}{n_0}\}\Big) \\ &\leq \sum_{i\in\mathcal{I}_0}\mathbb{P}\Big(\{p_i < \frac{\alpha}{n_0}\}\Big) \\ &\leq n_0\frac{\alpha}{n} \leq \alpha \end{aligned} \tag{3}$$

Hence the Bonferroni method controls the FWER under the level of α strongly, without additional assumption on the dependence on the hypotheses. This allows its application in many cases. However, it is very conservative, since the new treshhold value may be very small even for smaller values of n. This may result easily in accepting the null hypothesis even when it is false, i.e. the method increases the type II error and decreases the statistical power of the test significantly.

B. Holm Bonferroni

In this method the p-values are ordered starting from the most significant one

$$p_{(1)} \leq p_{(2)} \leq \cdots \leq p_{(n)} \tag{4}$$

and the associated hypothesis will be denoted by $H_{(1)}, H_{(2)}, ..., H_{(n)}$. At a significance level α, we look for the minimal index (k), s.t.

$$p_{(k)} > \frac{\alpha}{n-k+1}. \tag{5}$$

If such an index exists, then only the values $p_{(1)}, p_{(2)}, .., p_{(k-1)}$ will be considered significant and only the hypotheses $H_{(1)},, H_{(2)}, ..., H_{(k-1)}$ will be rejected. For $k = 1$ all the hypotheses are accepted, and if such an index k does not exist, all hypotheses are rejected. The adjusted p-values are given with

$$p_{(i)}^{(a)} = \min\{(n-i+1)p_i, 1\}$$

We will prove that the procedure controls the FWER strongly (for more details see [2]). Let us suppose that (i_0) is the first index in the ordering (4) defined above, such that the hypothesis $H_{(i_0)}$ is the first true hypothesis. Thus all $H_{(1)},, H_{(2)}, ..., H_{(i_0-1)}$ are false. The Holm-Bonferroni procedure will make a false rejection if $k > i_0$ and thus

$$p_{(i)} \leq \frac{\alpha}{n-i+1} \quad \text{for all} \quad i = \overline{1, i_0}$$

Let $N_0 = \{(i) | H_{(i)}$ is true hypothesis$\}$, $|N_0| = n_0$.

Then

$$n - (i_0) + 1 \geq n_0$$

i.e.

$$p_{(i_0)} \leq \frac{\alpha}{n-(i_0)+1} \leq \frac{\alpha}{n_0}.$$

Thus the probability of false rejection is

$$\begin{aligned} &\mathbb{P}\big(p_{(i_0)} \leq \frac{\alpha}{n-i_0+1}\big) \leq \mathbb{P}\big(p_{(i_0)} \leq \frac{\alpha}{n_0}\big) \\ &\leq \sum_{i\in N_0}\mathbb{P}\big(p_i \leq \frac{\alpha}{n_0}\big) \leq \alpha \end{aligned}$$

Thus, the Holm-Bonferroni procedure controls the FWER strongly. This is achieved under no assumption of independence of the testing hypothesis. The test is more powerful than Bonferroni. However, for large number of tests it will still result in low power.

The Holm-Bonferroni procedure is a so called step-down procedure, since it looks forward in the sequence (4). We stop as soon as we find a p-value which passes the corresponding threshold. Next we present a step-up-procedure proposed by Hochberg [7] - this type of procedures consider the sequence (4) backwards, looking for the highest indexed p-value which will pass a given threshold. These type of procedures in general have more power than step-down procedures.

C. Hochberg

Hochberg's method [7] is a step-up version of the Bonferonni test. In the sequence of p-values (4), we look for the largest index (k), s.t.

$$p_{(k)} \leq \frac{\alpha}{n-k+1} \tag{6}$$

All values below this index are considered significant.

This method controls FWER at the level α, under the assumption that the test statistics are independent. Alternative assumptions also provide control of FWER [8].

III. Methods that control the FDR

The cost of controlling FWER is high especially when we have to conduct large number of tests - all FWER controling procedures do not scale good with n. It has been shown that as $n \to \infty$, the number of rejected hypotheses goes to zero [**?**]. Even more, in exploratory studies in new fields, or genome-wide association studies few false positives are not seen as a bad thing since they may lead to more detailed research which may bring new knowledge. This lead Benjamini and Hochberg to introduce an alternative approach to the FWER control. In their paper [9] from 1995 they have defined the false discovery rate (FDR) and also introduced a procedure which controls this rate.

Based on Table 1, we give the following definitions of False discovery proportion and the False discovery rate

Definition 1: Based on Table I we define

- Family discovery proportion FDP

$$FDP = \frac{V}{max\{R,1\}} = \begin{cases} \frac{V}{R} & \text{if } R \geq 1 \\ 0 & \text{otherwise} \end{cases} \tag{7}$$

- False discovery rate FDR

$$FDR = \mathbb{E}(FDP) \tag{8}$$

In this section we will explain Benjamini-Hochberg method and its modification Benjamini-Yakutieli method. Both methods control FDR. For more details on these methods and other procedures we refer to [9], [10], [3], [11].

A. Benjamini-Hochberg

This procedure was proposed by Benjamini and Hochberg in 1995 [9]. As before, we consider the ordered sequence of p-values (4). For a chosen value q for FDR, we look for the maximum index k, s.t.

$$p_{(k)} < \frac{k \cdot q}{n} \tag{9}$$

Then only the hypothesis $H_{(1)}, H_{(2)}, ..., H_{(k)}$ will be rejected, i.e. all p-values $p_{(1)}, p_{(2)}, ..., p_{(k)}$ will be considered significant, even if they did not fulfill their corresponding criterium. The adjusted p-values are given with

$$p_{(i)}^{(a)} = \min\{\min_{j \geq i}\{\frac{np_{(j)}}{j}\}, 1\} \tag{10}$$

Benjamini and Hochberg have proved that, given that the test statistics are independent, their procedure controls the FDR at the chosen level q, i.e.

$$FDR \leq \frac{n_0}{n} q$$

Details of the proof can be also found in [3].

We remark here that the level q is adaptive. There is a discussion in the literature what is the appropriate choice of q and whether it should be fixed before we see the results from the tests, or after. However, it does depend on the desired statistical power and type I error, but the preferred values also change across research fields. Lower p-values may ask for lower q threshold . We can also use values as high as 0.2 if it is important to us not to miss potentially important findings. More details on the estimation of FDR can be found for ex. in [12], [13].

B. Benjamini - Yekutieli method

Benjamini and Yekutieli have shown that the previous method controls the FDR also when the test statistics have positive regression dependency on each of the test statistics corresponding to the true null hypotheses. They have suggested that this conditions are general enough to be applied in wide range of problems. They have proven the following theorem

Theorem 1: When the Benjamini Hochberg procedure is conducted with $\frac{q}{\sum_{i=1}^{n} \frac{1}{i}}$ taking the place of q in (9), it always controls the FDR at level less than or equal to $\frac{n_0}{n} q$.

Thus, for generalization we consider the following criterion

$$p_{(i)} \leq \frac{i}{n \cdot c(n)} q \tag{11}$$

If the tests are independent or positively correlated, then we choose $c(n) = 1$. Under arbitrary dependence we take $c(n) = \sum_{i=1}^{n} \frac{1}{i}$. The procedure controls the FDR under all assumptions but thus it may have lower power..

IV. Application to an ophthalmological study

We will apply the presented methods on a data from our original study "Ophthalmic manifestation in children and young adults with Down syndrome and congenital heart defects" [1]. The purpose of the study was to investigate whether different types of ophthalmic manifestations are associated with congenital heart diseases (CHD). The study included 185 subjects with Down syndrome from Caucasian population, divided in two groups - group with CHD with 51 subjects, and group without CHD with 134 subjects. We have conducted and reported results on multiple hypotheses tests, 30 in the whole study. We will consider here the family of tests of independence based on the two groups with and without CHD, with respect to different ophthalmic characteristics. The tests were conducted under confidence level $\alpha = 0.05$.

The presented methods are summarized in Table II. The originally obtained p-values are ordered and presented in the second column. For each index $i = \overline{1, 20}$ the threshold values for the value $p_{(i)}$ with respect to each of the presented methods are given in the corresponding column. We remark that we can also present the adjusted p-values, and this is easily done in R with the function p.adjust. For the details of the function we refer to the R library [14]. For more details on the methods of adjustment of p-values we refer to [4] and [15].

Without adjustment, the first three p-values are less than the confidence level $\alpha = 0.05$ and they were reported as statistically significant. However, based on the medical practice and previous research, they were not considered to be medically significant and no further recommendations were made. If we apply any of the presented methods, we would come to similar conclusions. Only under Benjamini-Hochberg procedure with high $q = 0.02$ the same three values would be on the border of being significant. All other adjustment did not give any significant p-values.

This however, does not diminish the importance of the study. There are only few published studies on the topic, and even without significant results, the study adds a lot of additional knowledge in the field. Our findings confirmed some of the results from previous studies on similar population, and as expected, there were differences, especially with the studies based on genetically different population (African and Asian). The study gave a lot of medical insight, so the statistical tests are not the main focus of our study. Still, the future researchers will benefit a lot from our data. One of the main benefits would be the calculation of the effect size based on our data. This parameter is used for estimation of the sample size for future studies, needed to achieve a certain power of the test (usually 0.8) [16]

TABLE II
ORIGINAL AND ADJUSTED P-VALUES FOR THE PRESENTED METHODS

In./C.	Original	Bonnferoni	HB	Hochberg	BH	BY
1	0.0095	0.0025	0.0025	0.0025	0.01	0.0028
2	0.0174	0.0025	0.0026	0.0026	0.02	0.0056
3	0.0244	0.0025	0.0028	0.0028	0.03	0.0083
4	0.0753	0.0025	0.0029	0.0029	0.04	0.0111
5	0.0872	0.0025	0.0031	0.0031	0.05	0.0139
6	0.3003	0.0025	0.0033	0.0033	0.06	0.0167
7	0.3098	0.0025	0.0036	0.0036	0.07	0.0195
8	0.3778	0.0025	0.0038	0.0038	0.08	0.0222
9	0.3804	0.0025	0.0042	0.0042	0.09	0.025
10	0.45	0.0025	0.0045	0.0045	0.1	0.0278
11	0.4753	0.0025	0.005	0.005	0.11	0.0306
12	0.5402	0.0025	0.0056	0.0056	0.12	0.0334
13	0.579	0.0025	0.0063	0.0063	0.13	0.0361
14	0.6632	0.0025	0.0071	0.0071	0.14	0.0389
15	0.6803	0.0025	0.0083	0.0083	0.15	0.0417
16	0.7151	0.0025	0.01	0.01	0.16	0.0445
17	0.8285	0.0025	0.0125	0.0125	0.17	0.0473
18	0.8874	0.0025	0.0167	0.0167	0.18	0.05
19	0.909	0.0025	0.025	0.025	0.19	0.0528
20	0.945	0.0025	0.05	0.05	0.2	0.0556

V. COMMENTS AND CONCLUSION

In our original paper, we have left the original p values with no adjustments for few reasons. Most of the previous relevant literature did not make any adjustments, and we wanted the results to be comparable. Many previous studies reported only the significant results and not the number of conducted tests (this is against the research protocol and may be a result of the so called p-hacking or p-harvesting). Also, the research topic is relatively new, so the field would benefit a lot from the new insight, as discussed above.

We believe authors should be aware of the consequences of multiple testing and there should be a standardized approach in presentation of the results. If p-values are to be presented, the authors should be able ro choose the right method, based on the type of the study and the desired power of the test. In some fields as genomics the adjustments are even mandatory [17]. In some fields, as in our study there is no need to adjust for multiple comparisons as long as all the p-values are listed for every comparison, and it is explicitly stated that multiple comparisons have been made. This will allow the reader to judge the results for himself.

Some authors also criticize the adjustment approach - they argue that it is difficult to choose the family of tests, and it is not possible to adjust across all the future tests done on the same data/fields. They add that different approaches may make the results incomparable. Many suggest to avoid p-values altogether and present the results through confidence intervals, effects sizes . Another option is doing Bayes analysis which incorporates prior knowledge of the problem and allows change of probabilities as new evidence arises. Avoiding p-values will avoid the so called drawer effect and may insure more reliable results [18].

References

[1] A. Ljubic, V. Trajkovski, M. Tesic, B. Tojtovska, and B. Stankovic, "Ophthalmic manifestations in children and young adults with down syndrome and congenital heart defects," *Ophthalmic Epidemiology*, vol. 22, no. 2, pp. 123–129, 2015, pMID: 25777312.

[2] E. Candes. (2018) Stanford university, course in theory of statistics, lecture 5. [Online]. Available: https://statweb.stanford.edu/~candes/stats300c/Lectures/Lecture5.pdf

[3] ——. (2018) Course in theory of statistics, lecture 7. [Online]. Available: https://statweb.stanford.edu/~candes/stats300c/Lectures/Lecture7.pdf

[4] J. P. Shaffer, "Multiple hypothesis testing," *Annual Review of Psychology*, vol. 46, no. 1, pp. 561–584, 1995.

[5] J. Goeman, R. Meijer, T. Krebs, and A. Solari, "Simultaneous control of all false discovery proportions in large-scale multiple hypothesis testing," 2016.

[6] J. Dunn and O. J. Dunn, "Multiple comparisons among means," *American Statistical Association*, pp. 52–64, 1961.

[7] Y. HOCHBERG, "A sharper Bonferroni procedure for multiple tests of significance," *Biometrika*, vol. 75, no. 4, pp. 800–802, 12 1988.

[8] S. K. Sarkar, "Some probability inequalities for ordered mtp_2 random variables: a proof of the simes conjecture," *Ann. Statist.*, vol. 26, no. 2, pp. 494–504, 04 1998. [Online]. Available: https://doi.org/10.1214/aos/1028144846

[9] Y. Benjamini and Y. Hochberg, "Controlling the false discovery rate: A practical and powerful approach to multiple testing," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 57, no. 1, p. 289–300, 1995.

[10] Y. Benjamini and D. Yekutieli, "The control of the false discovery rate in multiple testing under dependency," *Ann. Statist.*, vol. 29, no. 4, pp. 1165–1188, 08 2001.

[11] S. Lee and D. K. Lee, "What is the proper way to apply the multiple comparison test?" *Korean Journal of Anesthesiology*, vol. 71, no. 5, p. 353–360, 2018.

[12] J. D. Storey and R. Tibshirani, "Statistical significance for genomewide studies," *Proceedings of the National Academy of Sciences*, vol. 100, no. 16, pp. 9440–9445, 2003.

[13] J. D. Storey, "A direct approach to false discovery rates," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 64, no. 3, pp. 479–498, 2002.

[14] p. s. v. . R library, "R: Adjust p-values for multiple comparison," https://stat.ethz.ch/R-manual/R-devel/library/stats/html/p.adjust.html, [Online; accessed 10-April-2019].

[15] S. P. Wright, "Adjusted p-values for simultaneous inference," *Biometrics*, vol. 48, no. 4, pp. 1005–1013, 1992.

[16] J. Cohen, *Statistical Power Analysis for the Behavioral Sciences*. Taylor & Francis, 2013.

[17] T. Nichols and S. Hayasaka, *Statistical Methods in Medical Research*, p. 419–446, 2003.

[18] J. P. A. Ioannidis, "Why most published research findings are false," *PLoS Med*, vol. 2, no. 8, p. e124, 08 2005.

Network Embedding: An Overview

Nino Arsov
Faculty of Computer Science and Engineering
Ss. Cyril and Methodius University
Skopje, Macedonia
narsov@manu.edu.mk

Georgina Mirceva
Faculty of Computer Science and Engineering
Ss. Cyril and Methodius University
Skopje, Macedonia
georgina.mirceva@finki.ukim.mk

***Abstract*—Networks are one of the most powerful structures for modeling problems in the real world. Downstream machine learning tasks defined on networks have the potential to solve a variety of problems. With link prediction, for instance, one can predict whether two persons will become friends on a social network. Many machine learning algorithms, however, require that each input example is a real vector. Network embedding encompasses various methods for unsupervised, and sometimes supervised, learning of feature representations of nodes and links in a network. Typically, embedding methods are based on the assumption that the similarity between nodes in the network should be reflected in the learned feature representations. In this paper, we review significant contributions to network embedding in the last decade. In particular, we look at four methods: Spectral Clustering, DEEPWALK, Large-scale Information Network Embedding (LINE), and node2vec. We describe each method and list its advantages and shortcomings. In addition, we give examples of real-world machine learning problems on networks in which the embedding is critical in order to maximize the predictive performance of the machine learning task. Finally, we take a look at research trends and state-of-the art methods in the research on network embedding.**

***Index Terms*—networks, network embedding, unsupervised learning, latent feature representations**

I. Introduction

Today, we live in a world that is connected more than ever before. But even before the rise of the internet and availability of vast amounts of data, the world was connected in many different ways, such as social and professional acquaintances and interactions. Today, however, centralized sources of data residing on digital storage systems reveal even more ways in which entities connect and interact with each other. Typical examples include social networks and scientific collaboration, among others. All these interactions can be easily represented with networks, known formally as graphs. The area dealing with this is called *network science*. One of the best resources on network science is [1]. A graph is a discrete structure that can take many shapes and be of many different types. A graph consists of a set of nodes and links that connect pairs of nodes. Moreover, the links can be either directed or undirected and have weights that quantify the relationship between the pair of nodes they connect. It is a discrete structure studied in discrete mathematics. In general, many real-world scenarios can be modeled using networks.

In recent years, machine learning has taken network science to a different level where the focus is concentrated on prediction tasks. For instance, one may want to predict the existence of a link between a pair of nodes (also known as *link prediction*). Another example is automatic detection of communities within networks (also known as *community detection*). Furthermore, nodes and links in a network may have attributes that describe the entity represented with the particular node that a scientist would like to predict. All these are examples of *downstream machine learning tasks*. Machine learning algorithms use real-valued input vectors and outputs to learn a latent function that maps each input vector into an output. In machine learning, inputs are either classified, i.e. assigned one or more labels from a finite set of labels, known as classes, or inputs are mapped to a real number that represents some quantity such as a product price or a custom measure. The former case is known as *classification*, while the latter is referred to as a *regression* task. In link prediction, for instance, two real-valued vectors that represent a pair of nodes are passed on to a machine learning algorithm to predict the existence of a link between them. This is a binary classification problem, in which the output label is either 0 or 1, which indicates the link's existence. Another task is node classification in which each node is assigned a class label, or even a real-number. The latter is known as node regression.

Network embedding. Network embedding refers to the approach of learning latent low-dimensional feature representations for the nodes or links in a network. The basic principle is to learn encodings for the nodes in the network such that the similarity in the embedding space reflects the similarity in the network. The scope of node embedding is varying and applicable to all kinds of different graph types. The advantage of node embedding as a technique is that it does not require feature engineering by domain experts.

This paper is organized as follows: in Section II we introduce various definitions and preliminaries to network embedding, in Section III we briefly review four network embedding methods, in Section IV we describe three case studies of network embedding used in subsequent downstream tasks, borrowed from [2], in Section V we conclude the paper and talk about the latest state-of-the-art network embedding methods.

II. Preliminaries

In this section, we formally give several definitions required to introduce node embedding techniques. First, we define graphs as a discrete structure.

Definition 1. *Graph.*
A graph $G(V,E)$ is a collection that amounts to a set of nodes $V = \{v_1, \dots, v_n\}$, called nodes, and a set of edges $E = \{e_{ij} \,|\, 1 \le i,j \le n\} \subseteq V \times V$, called links. When G is an undirected graph, if $e_{ij} \in E$, then $e_{ji} \in E$ and vice versa, and when G is directed, $e_{ij} \in E$ does not necessarily imply that $e_{ji} \in E$.

Definition 2. *Weighted graph.*
A weighted graph $G(V,E,W)$ is a collection that amounts to a set of nodes $V = \{v_1, \dots, v_n\}$, a set of links $E = \{e_{ij} \,|\, 1 \le i,j \le n\} \subseteq V \times V$, and a set of weights $W = \{w_{ij} \,|\, 1 \le i,j \le n, w_{ij} \ge 0, w_{ij} \in \mathbb{R}_+\}$. If $e_{ij} \notin E$, then $w_{ij} = 0$, and otherwise $w_{ij} > 0$.

The neighborhood of a node is generated using a search strategy that traverses the graph, such as Breadth-First Search (BFS), Depth-First Search (DFS), or a random walk. The neighborhoods of different nodes can have different sizes and they can overlap, i.e., $N(u) \cap N(v) \neq \varnothing$.

We now proceed with definitions related to node embedding. A node embedding is also known as a feature vector or a feature representation. The dimensionality of the embedding is given by $d \ge 1$ and is usually assumed to be known before the learning process takes place.

Definition 3. *Node embedding.*
Let $d \ge 1$ be the dimensionality of the node embeddings. A node embedding function f is a map $f : V \longrightarrow \mathbb{R}^d$ maps each node $v \in V$ to a real-valued feature vector in $\mathbb{R}^d$.

III. Selected Node Embedding Methods

In this section we give a brief review of four prominent network embedding methods: *Spectral Clustering* [3], DeepWalk [4], Large-scale Information Network Embedding (*LINE*) [5], and node2vec [2]. They all have the same goal of learning optimal node embeddings for large networks that can later be used in any downstream machine learning task.

A. Spectral Clustering

Spectral Clustering is a matrix factorization approach to network embedding based on the Laplacian matrix of a graph G. Spectral clustering was originally proposed to address the problem of partitioning a graph into disjoint sets. Here, the edges of a graph can have weights, denoting the similarity between nodes. Intuitively, we want to find a partition of the graph, so that the edges between groups have a small weight and the edges within a group have a large weight. This is closely related to the minimum-cut problem. For two disjoint node sets $B, C \subset V$, the cut between B and C is defined as

$$Cut(B,C) = \sum_{v_i \in B, v_j \in C} A_{ij},$$

where A is the adjacency matrix of the graph. Any d-way partition $(C_1, C_2, \dots, C_d)$ should satisfy $\bigcup_{i=1}^{d} C_i = V$, and $C_i \cap C_j = \varnothing$ for all $i \neq j$.

In [3], spectral clustering was chosen to extract node representations in social networks due to its effectiveness in various domains and the availability of a huge number of existing linear algebra packages to help solve the problem.

To find a good k-way partition, the problem can be formulated as $\min cut(C_1, C_2, \dots, C_k) = \sum_{i=1}^{k} cut(C_i, V/C_i)$.

In practice, this formulation of the problem may return trivial partitions like a group consisting of only one node, separated from the rest of the network. There exist alternative objectives capable of finding a somehow "balanced" partitioning by additionally taking the group size into account [3]. One commonly used objective is the normalized cut

$$Ncut(C_1, \dots, C_k) = \frac{1}{k} \sum_{i=1}^{k} \frac{cut(C_i, V/C_i)}{vol(C_i)},$$

where $vol(C_i) = \sum_{v_j \in C_i} d_j$, and d_j is the degree of node v_j. Let

$$H_{ij} = \begin{cases} 1/\sqrt{vol(C_j)} & \text{if node } i \text{ belongs to community } C_j \\ 0 & \text{otherwise.} \end{cases} \tag{1}$$

Then

$$Ncut(C_1, C_2, \dots, C_k) = \frac{1}{k} Tr(H^T L H),$$

such that $L = D - A$ is the graph Laplacian. Considering that $H^T D H = I$, the Ncut problem can be rewritten as

$$\begin{aligned} &\min_{C_1,\dots,C_k} Tr(H^T L H) \\ &\text{s.t. } H^T D H = I \\ &H \text{ conforms to Equation (2)} \end{aligned}$$

If we define $S = D^{1/2} H$, the problem can be transformed to

$$\begin{aligned} &\min_{S} Tr(S^T \tilde{L} S) \\ &\quad \text{s.t. } S^T S = I, \end{aligned} \tag{2}$$

where $\tilde{L} = D^{-1/2} L D^{-1/2} = I - D^{-1/2} A D^{-1/2}$ is the normalized Laplacian [3].

The optimal solution of S corresponds to the first d eigenvectors of the normalized graph Laplacian $\tilde{L}$ with the smallest eigenvalues. Typically in spectral clustering, a post-processing step like k-means clustering is applied to S or H to find a disjoint partition [3].

In summary, given a network A, spectral clustering is done by constructing a normalized Laplacian $\tilde{L}$ and then computing the first (smallest) d eigenvectors as the social dimensions, i.e., the feature representations of the nodes in the network. Finally, spectral clustering selects the d smallest eigenvectors of the normalized Laplacian $\tilde{L}$ as node embeddings [3].

B. DeepWalk

In DeepWalk, deep learning (unsupervised feature learning) was used for the first time to learn social representations of a graph's nodes by modeling a stream of short random walks. The algorithm learns latent feature representations that encode social relations in a continuous vector space with a relatively small number of dimensions. DeepWalk generalizes neural language models to process a special language composed of a set of randomly-generated walks. These neural language models have been used to capture the semantic and syntactic structure of human language, and even logical analogies [4].

DeepWalk outperformed other latent representation methods for creating social dimensions, especially when the labelled nodes are scarce. The representations learned by DeepWalk make strong predictive performance with very simple linear models highly possible. In addition, the inferred representations are general and can be combined with any classification method [4]. DeepWalk is an online algorithm and is trivially parallelizable.

Problem definition. DeepWalk considers the problem of classifying members of a social network into one or more classes (categories). Let $G = (V, E)$ and let $G_L = (V, E, X, Y)$ be a partially labeled social network, with input features $X \in \mathbb{R}^{V \times S}$, where S is the size of the feature space for each attribute vector, and $Y \in \mathbb{R}^{|V| \times |\mathcal{Y}|}$ given that $\mathcal{Y}$ is the set of possible labels.

The goal of DeepWalk is to learn $X_E \in \mathbb{R}^{|V| \times d}$, where d is a small number of latent dimensions. These low-dimensional representations are distributed; meaning each social phenomena is expressed by a subset of the dimensions and each dimension contributes to a subset of the social concepts expressed by the space [4].

The method satisfies these requirements by learning representation for nodes from a stream of short random walks, using optimization techniques originally designed for language modeling [4].

DeepWalk. The algorithm consists of two main components; first a random walk generator, and second, an update procedure. The random walk generator takes a graph G and samples uniformly a random node v_i as the root of the random walk $\mathcal{W}_{v_i}$. A walk samples uniformly from the neighbors of the last node visited until the maximum length (t) is reached. While the length of the random walks in the experiments is fixed, there is no restriction for the random walks to be of the same length. These walks could have restarts (i.e., a teleport probability of returning back to their root), but the preliminary results have not shown any advantage of using restarts. In practice, the implementation specifies a number of random walks γ of length t to start at each node [4].

SkipGram. SkipGram is a language model that maximizes the co-occurrence probability among the words that appear within a window, w, in a sentence. It approximates the conditional probability using an independence assumption as the following

$$P(\{v_{i-w}, \dots, v_{i+w}\} v_i | \Phi(v_i)) = \prod_{j=i-w, j \neq i}^{i+w} P(v_j | \Phi(v_i)),$$

where $\Phi(v_i)$ is a feature representation vector for node v_i. The purpose of the SkipGram model in DeepWalk is to capture the local structure around the node v_i, defined the surrounding w nodes, i.e., neighbors.

Hierarchical softmax. Given that $u_k \in V$, calculating $P(u_k | \Phi(v_j))$ is not feasible. Computing the partition function to be used as a normalization factor is also expensive, and instead, DeepWalk resorts to the hierarchical softmax. The nodes are assigned to the leaves of a binary tree, turning the prediction problem into maximizing the probability of a specific path in the hierarchy. The process is shown in [4, Fig. 3]. This reduces the computational complexity of calculating $P(u_k | \Phi(v_j))$ from $O(|V|)$ to $O(log|V|)$.

The training process can be sped up further by assigning shorter paths to the frequent nodes in the random walk. Huffman coding could help reduce the access time of frequent elements in the tree [4].

Finally, the complete DeepWalk algorithm and SkipGram are given in [4].

C. LINE

LINE (short for Large-Scale Information Network Embedding) is a representation learning algorithm that learns an embedding model for real world information networks.

In practice, information networks can be either directed (e.g., citation networks) or undirected (e.g., social network of users in Facebook). The weights of the edges can be either binary or take any real value. Embedding an information network into a low-dimensional space is useful in a variety of applications. To conduct the embedding, the network structures must be preserved. The first intuition is that the local network structure, i.e., the local pairwise proximity between the nodes, must be preserved [5].

The first-order proximity between two nodes u and v is expressed by the edge weight w_{uv}. The second-order proximity is expressed by the similarity between p_u and p_v, where $p_u = (w_{u,1}, \dots, w_{u,|V|})$.

The LINE embedding model preserves both first- and second-order proximities.

LINE with First-order Proximity. The first-order proximity refers to the local pairwise proximity between the nodes in the network. To model the first-order proximity, for each undirected edge (i, j), we define the joint probability between node v_i and v_j as follows:$p_1(v_i, v_j) = 1/exp(-u_i^T \cdot u_j)$, where $u_i \in \mathbb{R}^d$ is the low-dimensional vector representation of node v_i. This defines a distribution $p(\cdot, \cdot)$ over the space $V \times V$, and its empirical probability can be defined as $\hat{p}_1(i, j) = w_{ij}$, where $W = \sum_{(i,j) \in E} w_{ij}$. To preserve the $W(i, j) \in E$ first-order proximity, a straightforward way is to minimize the following objective function based on Kullback-Leibler (KL) divergence is used for $d(\cdot, \cdot)$:

$$O_1 = -\sum_{(i,j)\in E} w_{ij} \log p_1(v_i, v_j). \tag{3}$$

The first-order proximity is only applicable for undirected graphs.

LINE with Second-order Proximity.

For each directed edge (i,j), the probability of context v_j generated by node v_i is defined as:

$$O_2 = \sum_{i\in V} \lambda_i d(\hat{p}_2(\cdot|v_i), p_2(\cdot|v_i)) \tag{4}$$

As the importance of the nodes in the network may be different, λ_i is introduced in the objective function to represent the prestige of vertex i in the network, which can be measured by the degree or estimated through algorithms such as PageRank. The empirical distribution $\hat{p}_2(\cdot|v_i)$ is defined as $\hat{p}_2(v_j|v_i) = \frac{w_{ij}}{d_i}$, where w_{ij} is the weight of the edge (i,j) and d_i is the out-degree of node i, i.e., $d_i = \sum_{k\in N(i)} w_{ik}$, $N(i)$ is the set of out-neighbors of v_i [5]. In the original paper, the authors set $\lambda_i = d_i$ and take advantage of KL-divergence as the distance function. Plugging KL-divergence into Equation (4) and setting $\lambda_i = d_i$ yields

$$O_2 = -\sum_{(i,j)\in E} w_{ij} \log p_2(v_j|v_i).$$

Combined first-order and second-order proximity was left as future work [5].

Negative sampling (model optimization). Optimizing objective O_2 is computationally expensive as it requires the summation over the entire set of nodes to calculate the conditional probability $p(\cdot|v_i)$. The authors address this issue by adopting the approach of negative sampling [6], which samples multiple negative edges according to some noisy distributions for each edge (i,j). The final objective function is

$$O_2 = \log \sigma(u'_j \cdot u_i) + \sum_{i=1}^{K} E_{v_n \sim P_n(v)} \left[\log \sigma(-u'_n \cdot u_i)\right],$$

where $\sigma(x) = 1/(1+exp(-x))$ is the sigmoid function and K is the number of negative edges. Moreover, the first term models the observed edges while the second term models the negative edges drawn from the noise distribution [5].

D. node2vec

node2vec is an algorithmic framework for learning continuous feature representations for nodes in networks. In node2vec, the goal is to learn a mapping of nodes to a low-dimensional space of features that maximizes the likelihood of preserving network neighborhoods of nodes. The authors have used flexible notions of a node's neighborhood and have designed a biased random walk procedure that sufficiently explores diverse neighborhoods. In its core, node2vec is a semi-supervised algorithm. The key characteristic of node2vec is its scalability as it scales to networks of millions of nodes [2].

Problem definition. Let $G = (V,E)$ be a given network. The node2vec framework is general and applies to any (un)directed, (un)weighted network [2]. Let $f : V \to \mathbb{R}^d$ be the mapping function from nodes to feature representations to be learned for future downstream tasks. Equivalently, f is a matrix of size $|V| \times d$ parameters. For every source node $u \in V$, the *network neighborhood* of node u generated through a neighborhood sampling strategy S is defined $N_S(u) \subset V$. The authors then extend the SkipGram architecture to networks.

In general, node2vec seeks to optimize an objective function that maximizes the log-probability of observing a network neighborhood $N_S(u)$ for a node u, conditioned on its feature representation, given by f:

$$\max_f \sum_{u\in V} \log P(N_S(u)|f(u)). \tag{5}$$

Since the problem given in the form above is intractable, the authors make two critical assumptions to make it tractable: conditional independence (the likelihood of observing a neighborhood node is independent of the likelihood of observing any other neighborhood node) and symmetry in the feature space (a source node u and any neighborhood node $n_i \in N_S(u)$ have a symmetric effect over each other). The likelihood is modeled using the softmax:

$P(n_i|f(u)) = exp(f(n_i) \cdot f(u)) / \sum_{v\in V} exp(f(v) \cdot f(u)$.

The objective in Equation (5) simplifies to

$$\max_f \sum_{u\in V} \left[-\log Z_u + \sum_{n_i \in N_S(u)} f(n_i) \cdot f(u) \right], \tag{6}$$

where $Z_u = \sum_{v\in V} exp(f(u) \cdot f(v))$ is the partition function for node u. Unfortunately, computing Z_u is computationally expensive for large networks, and therefore, following the LINE principles, Z_u is approximated with negative sampling [2]. The authors use stochastic gradient descent (SGD) to optimize Equation (6) and attain scalability for large networks.

Neighborhood generating strategies. The neighborhoods generated within node2vec are not restricted to just the immediate neighbors: the authors propose a randomized sampling strategy based on random walks to generate a diverse neighborhood [2]. The neighborhood size is constrained to k. BFS and DFS are two extreme sampling strategies: the first samples the immediate neighbors of a node u, while the latter samples nodes sequentially at increasing distances from the source node u. Nodes sampled with these two strategies are capable of conforming to the homophily hypothesis [7], [8] and structural equivalence [9]. More information can be found in the original paper [2]. Two parameters, p and q, are used to control the random walks. First, p controls the likelihood of immediately revisiting a node in the walk. Setting it to a high value makes the walk less incline to sample an already-visited node in the previous two steps. The parameter q differentiates between “inward” and “outward” nodes. If $q < 1$, the walk is more inclined to visit nodes further from t. This behavior is reflective of DFS. If $q > 1$, the random walk is biased towards nodes close to node t, which is reflective of BFS.

Finally, node2vec initializes r random walks per node to generate diverse sets of node neighborhoods. More details on the search strategy can be found in [2]. The node2vec algorithm is given in Algorithm 1, and the biased random walk procedure is given in [2].

Algorithm 1 NODE2VEC

```
LearnFeatures (Graph $G = (V, E, W)$, Dimensions $d$, Walks per node $r$, Walk length $l$, Context size $k$, Return $p$, In-out $q$)
$\pi$ = PreprocessModifiedWeights$(G, p, q)$
$G' = (V, E, \pi)$
Initialize $walks$ to Empty
for $iter = 1, \dots, r$ do
    for all nodes $u \in V$ do
        $walk$ = node2vecWalk$(G', u, l)$
        Append $walk$ to $walks$
$f$ = StochasticGradientDescent$(k, d, walks)$
return $f$
```

The three phases of node2vec, i.e., preprocessing to compute transition probabilities, random walk simulations and optimization using SGD, are executed sequentially. Each phase is parallelizable and executed asynchronously, contributing to the overall scalability of node2vec.

IV. Case Studies

A. Case Study 1: Les Misèrables Network

This section was borrowed from [2]. It studies the Les Misèrables network and four node embedding algorithms: Spectral Clustering, DeepWalk, LINE, and node2vec.

In this network, nodes correspond to characters in the novel Les Misèrables [10] and edges connect coappearing characters. The network has 77 nodes and 254 edges. The embedding dimension was set to $d = 16$ and node2vec was run to learn a feature representation for every node in the network. The feature representations are clustered using k-means and the nodes were colored according to the cluster assignments [2]. Fig. **??** (top) shows the example when $p = 1, q = 0.5$. Regions of the network are colored using the same color. In this setting node2vec discovers clusters/communities of characters that frequently interact with each other in the major sub-plots of the novel. This characterization closely relates with homophily [2]. In order to discover which nodes have the same structural roles the authors use the same network but set $p = 1, q = 2$. Here, node2vec obtains a complementary assignment of node to clusters such that the colors correspond to structural equivalence as illustrated in Fig. **??**(bottom). For instance, node2vec embeds blue-colored nodes close together. These nodes represent characters that act as bridges between different sub-plots of the novel. Similarly, the yellow nodes mostly represent characters that are at the periphery and have limited interactions. In [**?**]grover2016node2vec, the the top plot reflects homophily and the bottom plot represents structural equivalence.

TABLE I
Macro-F_1 Scores of different network embedding algorithms. This table is borrowed from the original node2vec paper [2].

Algorithm	Dataset		
	BlogCatalog	PPI	Wikipedia
Spectral Clustering	0.0405	0.0681	0.0395
DeepWalk	0.2110	0.1768	0.1274
LINE	0.0784	0.1447	0.1164
node2vec	**0.2581**	**0.1791**	**0.1552**
node2vec settings (p, q)	(0.25, 0.25)	(4,1)	(4, 0.5)

B. Case Study 2: Multilabel Classification

This section was also borrowed from [2]. The authors compare the Macro-F_1 scores for multi-label classification on three datasets: BlogCatalog, Protein-Protein Interactions (PPI), and Wikipedia (see [2]).

All these networks exhibit a fair mix of homophilic and structural equivalences [2].

The node feature representations were input to a one-vs-rest logistic regression classifier with L_2 regularization. The train and test data was split equally over 10 random instances. The authors used the Macro-F_1 scores for comparing performance, shown in Table IV-B.

C. Case Study 3: Link Prediction

In link prediction, the task is to predict the existence of links between pairs of nodes given a network with a certain fraction of edges removed. Interestingly, the authors of node2vec were the first to use the learned feature representations for link prediction. In this task, the challenge is to combine the features for each pair of nodes; usually, a pair-wise operator between $f(u)$ and $f(v)$, such as the average ($\frac{f(u)+f(v)}{2}$), the Hadamard, i.e., pair-wise product $((f(u) \circ f(v))_i = (f(u))_i(f(v))_i)$, the Weighted-L1, and Weighted-L2 norm is used to combine the features.

The link prediction performance of Spectral Clustering, DeepWalk, LINE, and node2vec was tested on two additional datasets along PPI: the Facebook dataset [11], in which nodes represent users and edges represent friendship relation between any two users, and the arXiv ASTRO-PH dataset [11], a collaboration network generated from papers submitted to arXiv where nodes represent scientists and edges represent collaboration, that is, an edge is present between two scientists if they have collaborated in a paper. With respect to Area Under Curve (AUC) scores on link prediction, node2vec again outperforms both DeepWalk and LINE with gain up to 3.8% and 6.5%, respectively. The scores are summarized in Table IV-C (borrowed from [2]). The authors reported that the Hadamard product is the most stable and gives the best performance across all networks on average [2].

V. Current Trends in Node Embedding Research

The four approaches that we took a look at in this paper form the base of network embedding. The most efficient

TABLE II
AUC Scores for Link Prediction of the Four Algorithms Using Different Operators to Embed Links: (a) Average, (b) Hadamard, (c) Weighted-L1 Norm, and (d) Weighted-L2 Norm, borrowed from [2]

Operator	**Algorithm**	**Dataset** Facebook	 PPI	 arXiv
(a)	Spectral Clustering	0.5960	0.6588	0.5812
	DeepWalk	0.7238	0.6923	0.7066
	LINE	0.7029	0.6330	0.6516
	node2vec	0.7266	0.7543	0.7221
(b)	Spectral Clustering	0.6192	0.4920	0.5740
	DeepWalk	**0.9680**	0.7441	0.9340
	LINE	0.9490	0.7249	0.8902
	node2vec	**0.9680**	**0.7719**	**0.9366**
(c)	Spectral Clustering	0.7200	0.6356	0.7099
	DeepWalk	0.9574	0.6026	0.8282
	LINE	0.9483	0.7024	0.8809
	node2vec	0.9602	0.6292	0.8468
(d)	Spectral Clustering	0.7107	0.6026	0.6765
	DeepWalk	0.9584	0.6118	0.8305
	LINE	0.9460	0.7106	0.8862
	node2vec	0.9606	0.6236	0.8477

method is node2vec [2] that easily outperforms the remaining three. During the last two years, however, there have been significant advances in developing novel embedding approaches applicable to various types of networks. The methods surveyed here are applicable to homogeneous networks, i.e., networks in which all nodes represent instances of the same entity. Network embedding in heterogeneous networks is more challenging and one of the methods that does this is metapath2vec [12]. Next, Modulized Non-Negative Matrix Factorization (N-NMF) learns representations that preserve the communities within the network [13]. For networks in which the nodes have multiple attributes (also known as attributed networks), one can use label-informed attributed network embedding [14]. A framework called struc2vec [15] learns embeddings that preserve the structural identity, which is a concept of symmetry in which network nodes are identified according to the network structure and their relationship to other nodes in a network. Nodes residing in different parts of a graph can have similar structural roles within their local network topology. This kind of embeddings can be learned via diffusion wavelets [16]. Node embedding can be extended to links that represent relationships in social networks, for instance—this is known as relationship embedding [17]. The following papers provide extensive surveys on network embedding: [18], [19].

VI. Conclusion

Network embedding is critical for applying machine learning approaches that are becoming ubiquitous in network science. In this paper, we reviewed four important network embedding techniques: spectral clustering, DeepWalk, LINE, and node2vec. The representations learned by node2vec manifest the best performance in downstream tasks. Learning disentangled representations is a popular research trend that tries to bring network embedding to a new level in which the black-box model is replaced by methods that learn representations in which each original feature is represented by one or more dimensions in the learned embeddings. Network embedding methods have contributed to a large extent in applying machine learning in network science.

References

[1] A.-L. Barabási *et al.*, *Network science*. Cambridge university press, 2016.

[2] A. Grover and J. Leskovec, "node2vec: Scalable feature learning for networks," in *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2016, pp. 855–864.

[3] L. Tang and H. Liu, "Leveraging social media networks for classification," *Data Mining and Knowledge Discovery*, vol. 23, no. 3, pp. 447–478, 2011.

[4] B. Perozzi, R. Al-Rfou, and S. Skiena, "Deepwalk: Online learning of social representations," in *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2014, pp. 701–710.

[5] J. Tang, M. Qu, M. Wang, M. Zhang, J. Yan, and Q. Mei, "Line: Large-scale information network embedding," in *Proceedings of the 24th international conference on world wide web*. International World Wide Web Conferences Steering Committee, 2015, pp. 1067–1077.

[6] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Advances in neural information processing systems*, 2013, pp. 3111–3119.

[7] S. Fortunato, "Community detection in graphs," *Physics reports*, vol. 486, no. 3-5, pp. 75–174, 2010.

[8] J. Yang and J. Leskovec, "Overlapping communities explain core–periphery organization of networks," *Proceedings of the IEEE*, vol. 102, no. 12, pp. 1892–1902, 2014.

[9] P. D. Hoff, A. E. Raftery, and M. S. Handcock, "Latent space approaches to social network analysis," *Journal of the american Statistical association*, vol. 97, no. 460, pp. 1090–1098, 2002.

[10] D. E. Knuth, *The Stanford GraphBase: a platform for combinatorial computing*. AcM Press New York, 1994, vol. 56.

[11] J. Leskovec and A. Krevl, "Snap datasets: Stanford large network dataset collection," http://snap.stanford.edu/data, april, 2019.

[12] Y. Dong, N. V. Chawla, and A. Swami, "metapath2vec: Scalable representation learning for heterogeneous networks," in *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining*. ACM, 2017, pp. 135–144.

[13] X. Wang, P. Cui, J. Wang, J. Pei, W. Zhu, and S. Yang, "Community preserving network embedding," in *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.

[14] X. Huang, J. Li, and X. Hu, "Label informed attributed network embedding," in *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining*. ACM, 2017, pp. 731–739.

[15] L. F. Ribeiro, P. H. Saverese, and D. R. Figueiredo, "struc2vec: Learning node representations from structural identity," in *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 2017, pp. 385–394.

[16] C. Donnat, M. Zitnik, D. Hallac, and J. Leskovec, "Learning structural node embeddings via diffusion wavelets," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. ACM, 2018, pp. 1320–1329.

[17] Y.-Y. Lai, J. Neville, and D. Goldwasser, "Transconv: Relationship embedding in social networks," 2019.

[18] P. Goyal and E. Ferrara, "Graph embedding techniques, applications, and performance: A survey," *Knowledge-Based Systems*, vol. 151, pp. 78–94, 2018.

[19] P. Cui, X. Wang, J. Pei, and W. Zhu, "A survey on network embedding," *IEEE Transactions on Knowledge and Data Engineering*, 2018.

Online sophistication of e-government services in North Macedonia

Izabela Akjimoska, Anita Antikj, Marjan Gusev

Ss. Cyril and Methodius University, Skopje, North Macedonia
iakimoska@gmail.com, anita.m.antic@gmail.com, marjan.gushev@finki.ukim.mk

***Abstract*—E-government sophistication was measured according to the EU commissions' 2010 benchmarks. This set of benchmarks is addressing 20 basic public services, 12 for citizen and 8 for businesses. Although a lot of EU countries have reached a very high performance of this benchmark in 2010, the performance in North Macedonia was less than the regional average on e-government. In this paper, we have scanned the overall achievement and concluded a growing trend for online sophistication of e-government services. The evaluation has shown that although there is a significant improvement, still the performance in 2018 is not on the level of the average performance of EU countries in 2010.**

***Index Terms*—E-government, online sophistication, full online availability**

I. Introduction

The research for sophistication of e-government services in North Macedonia become an important topic in the previous two decades, since the official measurement of online sophistication in European countries have not evaluated the situation in North Macedonia. According to this measurement, different levels of online sophistication in public services have been achieved throughout the years.

The objective of the paper is to overview the present development of e-government service sophistication in North Macedonia and conclude what is the level of development in correlation with other European countries, as well as presenting the full online availability of those public services.

The first research question is regarding the evolution of online sophistication and current situation in North Macedonia, and the second research question addresses the comparison of e-government online sophistication to other EU countries.

The methodology used consists of the EU e-government benchmarks set by the European Commission and monitored by Capgemini for online sophistication measurement of 20 public online services [1], [2].

The research aims to present the state in which the public services are developing and their importance to the development of the business community, the integration between the state administrator and the citizens, as well as, the integration within a specific governmental institution.

The paper follows the next structure. Section II explains the used methods in this research. Results are presented in Section III and discussed in Section IV. Conclusion and future work directions are elaborated in Section V.

II. Methods

According to the set of benchmarks, the framework consists of the evaluation of online sophistication for the following 20 basic public services, given in Table 1.

TABLE I
The 20 public services (EU e-Government benchmarks)

	Citizen		Business
1	Income taxes	13	Social contributions
2	Job search	14	Corporate tax
3	Social Security Benefits	15	VAT
4	Personal documents	16	Registration of a new company
5	Car registration	17	Submission of data to statistical offices
6	Building permission	18	Customs declaration
7	Declaration to the police	19	Environment-related permits
8	Public libraries	20	Public procurement
9	Certificates		
10	Enrollment in higher education		
11	Announcement of moving		
12	Health related services		

Each service can fall in one of the two target groups: citizens or business. In addition, there are four clusters that further divide the 20 basic services, named as income cluster, registration cluster, returns cluster, and permits cluster, presented later in the paper.

Some services have sub-services like the service Social Security Benefits which is made of the following sub-services: unemployment benefits, child allowances, medical costs, and student grants. The Personal documents service consists of: ID, Passport, and drivers license. Certificates are consisted of: birth, marriage, and death certificates.

Each elementary service or sub-service is graded on a scale from 0 to 5. Till 2007 each service could have had the maximum grade of 4, but in 2007 the European Commission introduced a new 5th stage which refers to the personalization of services. Grade 0 is interpreted as no information available online; 1 is interpreted as relevant online information available; 2 is interpreted as one way interaction; 3 as two way interaction; 4 as transaction fully available online, or 5 is that the service is targeted, proactive with an automated service delivery. [3], [4] The online sophistication for an elementary service is calculated in percentages as the ratio between the grade and the maximum attainable grade. The final online sophistication level is the average of the sophistication of the 20 basic services.

The fully online sophistication is another parameter that shows the percentage of all fully sophisticated services [1]. Note that some of services reach maximum grade of 4 and others 5, which determines that this parameter is essential in getting an overall indication of e-government services performance.

To present the evolution and growth of e-government services, we have included previous measurements reported in corresponding published papers.

III. Results

A. Online sophistication

- Income Taxes

Year	Stage of sophistication	Max. Stage
2018	5	5

The service provider for this service is The Public Revenue Office. Its site provides easy access to the latest information about the activities and services as well as the taxation in the country. The tax information are adjusted to different types of clients with different levels of knowledge and needs. It offers a possibility for downloading the appropriate forms in order to start the procedure of tax-declaration. The process of paying income taxes can be handled completely electronic. - Website: www.ujp.gov.mk
- Online sophistication: 100%

- Job search

Year	Stage of sophistication	Max. Stage
2018	3	4

This service is offered by the Employment Service Agency of North Macedonia which is a public institution that carries out professional, organizational, administrative and other activities related to employment and unemployment insurance, and provides support, assistance and services for the participants in the labor market. They introduced an e-Work system: e-rabota.avrm.gov.mk which is divided for different users:

- Individual: allowed to search through the database of job offerings according to ones work profile, or list a history of previous employments (registered labor relations). In order to use this feature, the person has to be registered in the system of the service provider.

- Legal Entity: can publish a free working place, register the establishment of a working relationship, or sign of a working agreement. For using this feature, owning a digital certificate is required.

- Temporary Employment Agency: same procedure as the previously explained feature

Services for other employers: used for reporting/removing employers to their clients. One digital certificate is needed, but each client must be registered separately in the system. - Website:www.avrm.gov.mk
- Online sophistication: 75%

- Social Security Benefits

Year	Stage of sophistication	Max. Stage
2018	3	5

This service is consisted of four sub-services including:

- Unemployment Benefits: provided by The Ministry of Labor and Social Policy. There is only a place explaining which documents are required www.mtsp.gov.mk, graded: 2

- Child allowances: provided by The Ministry of Labor and Social Policy. It is stated which documents are required and there is a possibility of downloading the appropriate forms from their site www.mtsp.gov.mk, but the rest of the procedure continues offline, graded: 3

- Medical costs reimbursement: provided by Health Insurance Fund. On their application najava.fzo.org.mk there is an option to apply for a refund of resources dependent on few medical basis, graded: 4

- Student grants: provided by the Ministry of Education and Science. A student applies for a scholarship/grant online and proceeds with the rest of the procedure offline konkursi.mon.gov.mk, graded: 3
- Online sophistication: 60%

- Personal documents

Year	Stage of sophistication	Max. Stage
2018	4	5

The service provider is the Ministry of Interior. As personal documents are considered to be Passport, ID, and drivers license. The site provides: scheduling a service, review of reservations, completed personal documents, and details about of the needed documents separately.
- Website: termin.mvr.gov.mk
- Online sophistication: 80%

- Car registration

Year	Stage of sophistication	Max. Stage
2018	2	4

It is provided by the Ministry of Interior. On the site there is only an explanation of the procedure for applying as well as a possibility to download the needed forms for registration.
- Website: mvr.gov.mk
- Online sophistication: 50%

- Building permission

Year	Stage of sophistication	Max. Stage
2018	4	4

The service provider is the Ministry of transport and communications in cooperation with the Association of the units of local self-government of North Macedonia - ZELS. The information system enables the conduct of procedures in all its phases only electronically, starting from the electronic submission and signing of the necessary documentation, electronic preparation and adoption of the acts in the procedure, electronic notification and provision of necessary data, as well

as, documentation from other involved entities and electronic issuance of acts in the procedure. For each phase the citizen will be notified promptly via an automated and independent email delivery system and a notification sent via the Automatic Short Message Service (SMS).
- Website: www.gradezna-dozvola.mk
- Online sophistication: 100%

- Declaration to the police

Year	Stage of sophistication	Max. Stage
2018	2	3

The provider of the service is the Ministry of Interior. Informative documents can be found on the Ministry's website, as well as some one-way interaction forms, like the online reporting of suspicious act or person, online reporting of child abuse (Red Button), an email form for initiating a traffic control patrol, and a list of Found and Lost items and their whereabouts in local police stations. One fully online procedure is the Reporting of the place of sojourn/residence or change of address for foreigners.
- Website: mvr.gov.mk
- Online sophistication: 66,66%

- Public libraries

Year	Stage of sophistication	Max. Stage
2018	4	5

Online access to search tools regarding 40 libraries in the country, some items are provided with their source on the Internet where the user can download in electronic form. Also, the user can book (reserve in advance) selected items in libraries where the lending procedure is automated with COBISS Lending software, although this is not available in all the libraries.
- Website: www.vbm.mk
- Online sophistication: 80%

- Certificates

Year	Stage of sophistication	Max. Stage
2018	4	4

Certificates are consisted of: birth, marriage, and death certificates. The provider of the public service is the Ministry of Justice in cooperation with the Office for management of registers. Each citizen can personally order a certificate online, pay online and receive the certificate via post office service.
- Website: e-portal.uvmk.gov.mk
- Online sophistication: 100 %

- Enrollment in higher education

Year	Stage of sophistication	Max. Stage
2018	2	4

This service is consisted of 4 sub-services including: Enrollment of new student: the enrollment is initiated online, but the documents have to be presented to a Student affairs office www.upisi.ukim.mk graded: 2 Enrollment in higher educational year/repetition of a year: the enrollment is initiated online, but the documents have to be presented to a Student affairs office www.iknow.ukim.mk graded: 2 Registering for an exam: the registration is initiated online, but a paper copy has to be submitted to a Student affairs office www.iknow.ukim.mk graded: 2 Certificate issuing: the issuing is initiated online, but a paper copy has to be received from a Student affairs office www.iknow.ukim.mk graded: 2 Student grants: the student applies for a scholarship/grant online and proceed with the rest of the procedure offline konkursi.mon.gov.mk graded: 2
- Online sophistication: 50%

- Announcement of moving

Year	Stage of sophistication	Max. Stage
2018	2	4

This public service is provided by the Ministry of Interior and refers to the change of a address within the country. The forms can be found on the webpage, no further line of the procedure is detected online.
- Website: mvr.gov.mk
- Online sophistication: 50%

- Health related services

Year	Stage of sophistication	Max. Stage
2018	2	4

Service provided by the Ministry of Health. Health related services include: Interactive consultation of available services: no information located, graded: 0 Interactive appointments: online service for appointments mojtermin.mk. graded: 4
- Online sophistication: 50%

- Social contributions

Year	Stage of sophistication	Max. Stage
2018	3	4

The service provider for this service is the Pension and Disability Fund of North Macedonia. The portal for e-Services that they offer uslugi.piom.com.mk. enables a complete electronic management for treating the declaration of social contributions.
- Website: www.piom.com.mk
- Online sophistication: 75%

- Corporate tax

Year	Stage of sophistication	Max. Stage
2018	4	4

This service is offered by The Public Revenue Office. As a part of USAIDs eGov initiative, taxes handling became completely electronic, allowing these features: submit a tax return, review and payment of tax, review and payment of a broadcasting fee, Registering a GPRS fiscal device, issuing certificates and other services, report irregularities, e-auctions; The process of paying corporate tax is can be managed electronically.
- Website: etax.ujp.gov.mk
- Online sophistication: 100%

- VAT

Year	Stage of sophistication	Max. Stage
2018	4	4

The service provider is The Public Revenue Office. This service is handled as a part of the previously explained service.
- Website: etax.ujp.gov.mk
- Online sophistication: 100%

- Registration of a new company

Year	Stage of sophistication	Max. Stage
2018	4	4

Provided by the Central Registry. It offers downloading the needed forms, as well as checking a legal subjects existence in the name database and its status before entering it in the system as a valid one. There is an option for electronic registration of a firm (depending what kind of a legal subject it is) that goes through the whole procedure electronically.
- Website: e-submit.crm.com.mk
- Online sophistication: 100%

- Submission of data to statistical offices

Year	Stage of sophistication	Max. Stage
2018	3	5

The State Statistical Office as a provider of this service, started an electronic data collection with a gradual introduction of e-forms for statistical surveys few years ago. For that purpose is created: eStat, which is intended for collecting data from business entities via an e-form. Half of the services are offered completely electronic. For these ones, the application is entered with a username and password, which are submitted to the responsible people of the business entities - data providers for individual surveys carried out by the State Statistical Office, graded: 5 The other half of the services (for example: workforce statistics) needs to be done on the traditional way; only the information is provided, graded: 1
- Website: estat.stat.gov.mk
- Online sophistication: 60%

- Customs declaration

Year	Stage of sophistication	Max. Stage
2018	4	5

The service e-Customs is provided by the Ministry of Finance with the cooperation of the The Customs Administration. In order to create an account on the e-Customs portal, the business entity has to provide documentation that can be concluded by a digital certificate. The portal provides support for processing declarations and excise documents, real-time tracking of the businesss goods exiting from the territory of the Republic of Macedonia, getting permits for import, export or transit, and submitting various types of requests to the Customs Administration in electronic form.
- Website: www.customs.gov.mk
- Online sophistication: 100%

- Environment-related permits

Year	Stage of sophistication	Max. Stage
2018	2	5

The website of the The Ministry of environment and physical planning has forms that can be downloaded and information regarding the service.
- Website: uslugi.gov.mk
- Online sophistication: 40%

- Public procurement

Year	Stage of sophistication	Max. Stage
2018	4	4

The Bureau for Public procurement has an online system for conducting public procurements and proceed e-auctions fully online.
- Website: e-nabavki.gov.mk
- Online sophistication: 100%

B. *Full online availability*

Full online availability is the extent to which there is fully automated and proactive delivery of the 20 key public services. It addresses the question of - Is it the service fully enabled? It is derived from the preceding section, which evaluates the quality of e-government service provision in a country, across the representative basket of 20 services. North Macedonia reaches the score of 62% in 2018.

IV. Discussion

In this section, we analyze the overall achieved state from the evaluation process, and compare it to the state the other EU countries have reached.

A. *Growth of e-Government sophistication in North Macedonia*

The United Nations 2018 Survey highlights a persistent positive global trend towards higher levels of e-government development. [5] For the first time in 2018 the main contributor of The Environmental Data & Governance Initiative scores improvement in all income groups is development of online services, suggesting that globally, there was a steady progress in improving e-government and public services provision online. [6]

In regards of the progress of e-Government sophistication of public services, North Macedonia is surrounded by good practices in developed countries, as well as in fellow neighbors' governments. These practices provide a guide to how public services can strive towards growth in the near future, because growth in Macedonia's' e-government sophistication has been detected since 2001 in research papers, but not so significant as other e-government's efforts. [7], [8]

Concerning regional progress, in a study made by the Regional School of Public Administration (taking into consideration Albania, Bosnia and Herzegovina, Kosovo, North Macedonia, Montenegro and Serbia), North Macedonia is underperforming on e-government and scores less well than the regional average. Nevertheless, earlier achievements in e-government initiatives in Macedonia contributed to an exceptional performance as a leader on transparency and a good performance, even though in overall the country is behind other countries on e-government in the past few years. [9]

The first measurement in a 2004 study has reported that the average online sophistication in North Macedonia is 9%, in

March 2006 the average increased to 32.75%, and in March 2007 it was 50%. The 2018 average score is 77% by 2010 benchmarks, so a significant increase is detected in all public services, represented in Fig. 1

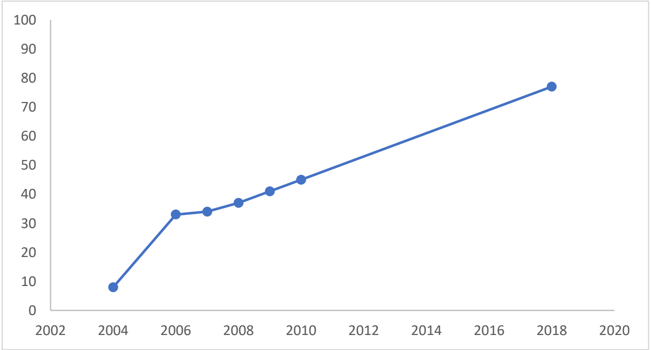


Fig. 1. Online sophistication growth in North Macedonia 2004-2018

In Fig. 2 is represented the fully online availability growth in North Macedonia for the period of 2004-2018. As we can see, until 2007 the result was 0, when it started growing from 10% so that in 2018 it reaches 62%. The sources for Fig. 1 and Fig 2. are included in the references bellow as they represent previous measurements for the same public services.

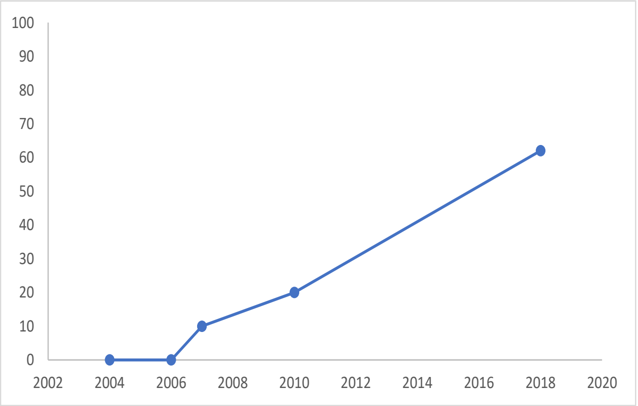


Fig. 2. Fully online availability growth in North Macedonia 2004-2018

B. Comparison to other EU countries

European countries lead e-government development globally, eight of the 11 new countries that joined the very-high performing group in 2018 are from Europe (Belarus, Greece, Liechtenstein, Malta, Monaco, Poland, Portugal and the Russian Federation).

The EU27+ average score for online sophistication in **2010** was 90%, measured by the 2010 benchmark methodology. [10] The average score in 2010 for Cyprus (71%), Romania (73%), Bulgaria (77%), Croatia (78%), Iceland (79%), correlate closely with the average score for North Macedonia for **2018**, measured by the same benchmarks and method, shown in Fig. 3.

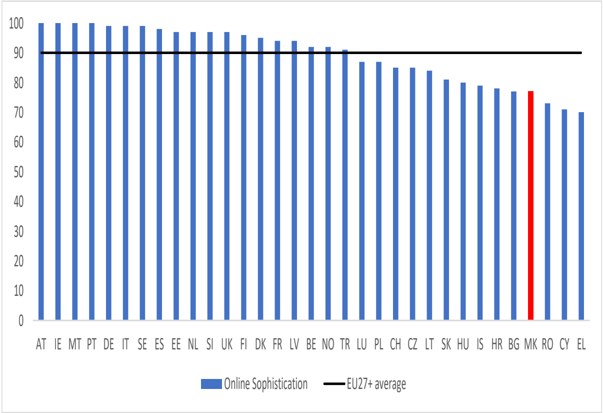


Fig. 3. Comparison of e-governement online sophistication between Macedonia in 2018 and EU27+ in 2010

On Full Online Availability (measured introducing a threshold to the 5stage maturity model which is mostly above the fourth or fifth sophistication level, depending on the service in question), the EU27+ average reaches 82% in 2010. The benchmark reveals that in Italy, Malta, Austria, Portugal and Sweden all 20 services are now 100% enabled. North Macedonia's average score is 62%, as shown in Fig. 4 compared to the scores of the other EU countries.

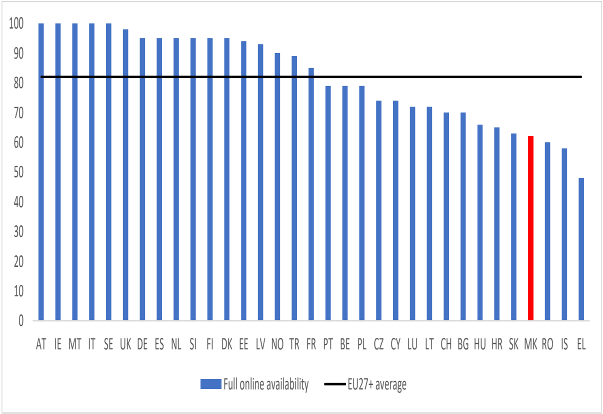


Fig. 4. Comparison of e-governement full availability between Macedonia in 2018 and EU27+ in 2010

In order to follow the steps that the European Union members take, the Ministry of Information Society and Administration of the Republic of Macedonia developed a National Strategy for e-Government 2010-2012 that addresses strategic approach in the utilization of information and communication technologies for more efficient operation of the state administration, but recent 2018 strategies impact only the empowerment of reforms in state administration. [11] This presents a setback in future development because there is a lack of strategic approach regarding ICT used in public institutions.

However, the focus on ICT is not crucial for success since it is only the enabler of e-government and is not the only important aspect that needs to be addressed. The focal point is breaking consumer resistance to using e-services and gaining their trust regarding security. That way encouraging more citizens, and businesses alike, to participate in public services online and forming a more open government that strives growth. That provokes the necessity of a long-term strategy for the digital society that should be developed and adopted to serve all different areas of e-government public services. [12]

V. Conclusion

E-governments benefits are enormous. If implemented correctly, it can transform the way that citizens access and interact with government and it can enable cooperation between independent agencies.

As shown in this paper, according to the result of the measured results of online available public services (which is 77% by 2010 benchmarks), a growing trend has been achieved in absolute values for online sophistication of e-government services in North Macedonia for the period of 2018.

The full online availability score for North Macedonia for 2018 is 62% which shows increasing values compared to the previous period of measurements.

The evaluated performance of e-government services compared to the accomplishments that it reached in the past years, it shows improvements. However, the achieved status in 2018 is lower than the average reached by EU countries in 2010. This implies a conclusion that there is a need for major changes and improvements, in order to catch up with the development of other countries, including high investments and strategic actions.

References

[1] Capgemini. (2006, June) Online availability of public services: How is Europe progressing? Web based survey on electronic public services report of the 6th measurement. [Online]. Available: http://ec.europa.eu/information_society/eeurope/i2010/docs/benchmarking/online_availability_2006.pdf

[2] Capgemini, R. Europe, IDC, Sogeti, and DTI. (2009, Nov) Smarter, Faster, Better eGovernment: 8th Benchmark Measurement. [Online]. Available: http://ec.europa.eu/information_society/eeurope/i2010/docs/benchmarking/egov_benchmark_2009.pdf

[3] M. Gusev, D. Spasov, and G. Armenski, "Growth of eGovernment Services in Macedonia (online sophistication of egovernment services)." *Informatica (Slovenia)*, pp. 397–406, 2007.

[4] M. Gusev and G. Armenski. (2006, Apr) Gap Analysis of eGovernment in Western Balkans. [Online]. Available: http://www.metamorphosis.org.mk

[5] United Nations. Gearing E-Government to support transformation towards sustainable and resilient societies. [Online]. Available: https://publicadministration.un.org/egovkb/Portals/egovkb/Documents/un/2018-Survey/E-Government%20Survey%202018_FINAL%20for%20web.pdf

[6] ——. United Nations E-Government Survey Results. [Online]. Available: https://publicadministration.un.org/egovkb/en-us/Reports/UN-E-Government-Survey-2018

[7] M. Gusev, M. Kostoska, and K. Kjirovski, "eGovernment Growth in Macedonia 2010," UKIM FINKI, Tech. Rep. II-2010-10, May 2010.

[8] K. Kiroski, M. Gusev, M. Kostoska, and S. Ristov, "Growth rate analysis of e-Government development 2012," in *Proceedings of the Fifth Balkan Conference in Informatics*, 2012. [Online]. Available: http://dl.acm.org/citation.cfm?id=2371316

[9] J. Millard, L. Thomasen, F. Kruja, F. Curcic, D. Elshani, G. Armenski, A. Drakcic, and B. Cvetkovic. (2015) E-Government Analysis: From E- to Open Government, Regional School of Public Administration. [Online]. Available: https://www.respaweb.eu/download/doc/eGov+-+From+E-Government+to+Open+Government.pdf/d3ab1cd43fa4cd3071be9cea7e4b0cd3.pdf

[10] Capgemini, IDC, "Digitizing public services in europe: Putting ambition into action: 9th benchmark measurement," *European Commission, Directorate General for Information Society and Media, Bruxelles*, 2010.

[11] Ministry of Information Society and Administration. National Strategy for e-Government 2010-2012. [Online]. Available: http://www.mioa.gov.mk/sites/default/files/pbl_files/documents/strategies/Strategija_za_e-Vlada-05.03.2010.pdf

[12] i2010 High Level Group. Benchmarking Digital Europe 2011-2015 A conceptual framework. [Online]. Available: https://joinup.ec.europa.eu/sites/default/files/document/2014-12/Benchmarking%20Digital%20Europe%202011-2015%20-%20A%20conceptual%20framework.pdf

Parallel Decoding with MAX-Log-MAP algorithm

Dejan Spasov
Faculty of Computer Science and Engineering
Sts. Cyril and Methodius University
Skopje, Macedonia
dejan.spasov@finki.ukim.mk

***Abstract*—Two or more parallelly concatenated recursive systematic convolutional codes are known as Turbo Codes. To improve thed speed of decoding, turbo codes may sub-optimally be decoded with serially-coupled MAX-Log-MAP decoders, where each MAX-Log-MAP decoder decodes one recursive systematic convolutional code. We propose a turbo decoding algorithm that, on a logical level of abstraction, is made of several turbo decoders working in parallel. Each turbo decoder is initialized with different recursive convolutional code. Practical implementation of the proposed algorithm may be achieved with a single turbo decoder, where MAX-Log-MAP decoders are working concurrently.**

Keywords—* *Turbo codes; MAX-Log-MAP decoding; parallel turbo decoding; convolutional codes

I. Introduction

Turbo Codes are a class of forward error correction codes invented by C. Berrou and first published in [1]. At the time of invention, turbo codes were the first practical system that achieved signal-to-noise ratio of 0.7 dB above the Shannon's limit while providing bit error probability of 10^{-5}[1]. In general, a turbo encoder may be considered any combination of two or more preferably identical (recursive) convolutional encoders connected via interleavers. Traditionally, a turbo code comprises two recursive systematic convolutional (RSC) encoders coupled in a parallel concatenation scheme (fig. 1).

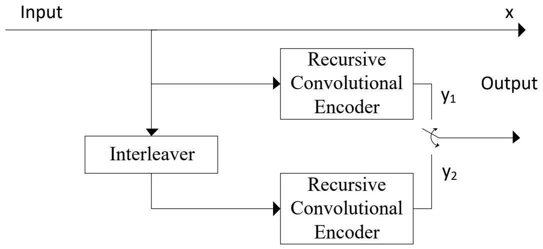

Input
x
Recursive Convolutional Encoder
y1
Output
Interleaver
y2
Recursive Convolutional Encoder

Fig. 1. Turbo encoder

Turbo codes are systematic codes, which means that the input sequence appears unmodified at the output as sequence. Fig. 1 shows two recursive systematic convolutional encoders that output two coded sequences (x, y_1) and (x, y_2), where sequences y_1 and y_2 represent the parity bits. The turbo code on fig. 1 initially has a code rate of 1/3; however, higher code rates may be achieved by applying various puncturing patterns. An example of puncturing pattern may be alternating between the parity bits y_1 and y_2. The interleaver is a device that pseudo-randomly permutes the input sequence. By providing random permutation on the input sequence, the interleaver allows identical recursive encoders to be used in the hardware design. Thus, two identical recursive systematic convolutional codes coupled with a random interleaver behave as two different recursive systematic convolutional codes.

Each recursive systematic convolutional code may be decoded with the MAX-Log-MAP algorithm [2]. Decoding of the turbo codes, which are made of two parallel systematic recursive convolutional codes, involves separate decoding of each of the systematic recursive convolutional codes.

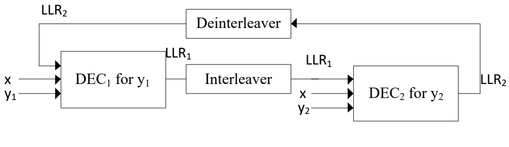


Fig. 2. Turbo decoder

Turbo decoder (fig. 2) is made of two MAX-Log-MAP decoders DEC$_1$ and DEC$_2$, which may share information several times before the turbo decoder outputs estimates for each bit. This information sharing is known as iteration. Thus, the turbo decoder performs several iterations before outputting decision for each bit. The turbo decoder may be configured to perform two iterations simultaneously [4]. For example, when DEC$_1$ is working on the i-th iteration, DEC$_2$ may be working on the (i-1)-th iteration. The two decoders DEC$_1$ and DEC$_2$ may be Max-Log-Map decoders [6], [7], which is a simplified variant of the MAP decoder that involves the Viterbi algorithm [2]. More on implementation issues may be found in [5] and [8]. Decoders DEC$_1$ and DEC$_2$ are configured to output an estimate of the logarithm of likelihood ratio (LLR) for each bit x_k

$$LLR(x_k) = \log \frac{\Pr\{x_k = 1|observation\}}{\Pr\{x_k = 0|observation\}} \qquad (1)$$

where $\Pr\{x_k = 1 or\ 0|observation\}$ is a posteriori probability of the bit x_k. The decoder DEC$_1$ is activated first and it decodes the encoded sequence (x, y_1) and outputs LLR$_1$ quantities for each bit x_k. The LLR$_1$ quantities are then fed to DEC$_2$ that decodes the encoded sequence (x, y_2) to produce its own estimates LLR$_2$. The turbo decoding process continues in iterative fashion, and in the next iteration LLR$_2$ quantities are fed into DEC$_1$.

From fig. 2, it may be observed that the turbo decoder is initialized with the first recursive convolutional code (x, y_1). In this paper we propose a turbo decoding scheme made of two serial turbo decoders that operate in parallel. The first turbo decoder is initialized with the first recursive convolutional code (x, y_1) and the second turbo decoder is initialized with the second recursive convolutional code (x, y_2). Results from both decoders are combined to produce the logarithm of likelihood ratio (LLR) for each bit x_k.

Funded by a project from the Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University.

II. THE MAX-LOG-MAP ALGORITHM

A convolutional encoder with M registers is finite state machine with 2^M states. Trellis diagram is labelled n-partite graph, in which every path represents a valid codeword (fig. 3). Vertices of the $n+1$ disjoint sets in the trellis represent all possible 2^M states of the encoder. Vertices are labelled as decimal numbers, such that the content of the leftmost register corresponds to the most significant bit in the decimal number. Edge labels represent the input letters to the encoder and the appropriate output letters produced by the encoder separated by the slash symbol.

Trellis diagram of convolutional codes gives a hint about the decoding process; if the received sequence does not represent a valid path through the trellis diagram, then we can conclude that errors have occurred. The decoding objective is to find the most probable valid path though the trellis. Several decoding algorithms exist for decoding convolutional codes. The most famous are the Viterbi algorithm [2] and the BCJR algorithm [3]. The Viterbi algorithm is universally used and is highly parallelizable. However, the applicability of the BCJR algorithm is limited. Instead of the BCJR algorithm, a sub-optimal MAX-Log-MAP is used in practical applications.

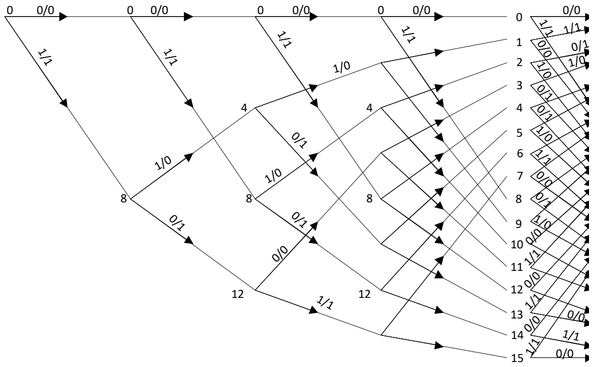


Fig. 3. Trellis diagram of a convolutional code

The MAX-Log-MAP algorithm can be envisioned as two stage process. In the first stage, known as *the forward α recursion*, the decoder moves through the trellis in left to right fashion and with each state s it associates a probability function $\alpha_i(s)$ that is recurrently computed. In the second stage, known as the *backward β recursion*, the MAX-Log-MAP decoder recurrently computes additional probability function $\beta_i(s)$ and then using the stored $\alpha_i(s)$ outputs the logarithm of likelihood ratio for each bit x_k.

Let $\alpha_i(s), s = 0,1,\dots,M-1$, be a set of state metrics on the n-partite trellis at the time i. The computation of the forward α probabilities starts from the initial conditions

$$\begin{cases} \alpha_0(s) = 0 & s = 0 \\ \alpha_0(s) = -\infty & s \neq 0 \end{cases} \tag{2}$$

and following the edges of the trellis with non-zero branch probabilities $\gamma_i(s,s')$, the decoder, at each iteration stores all $\alpha_i(s)$ and computes $\alpha_{i+1}(s)$, according to

$$\alpha_{i+1}(s') = \max_s\big(\alpha_i(s) + \gamma_i(s,s')\big) \tag{1}$$

where $s' = 0,1,\dots,M-1$. Stored $\alpha_i(s)$ are used during the backward computation in order to compute *the log-likelihood-ratio* (1) for the *i*-th information bit x_i.

Let $\beta_i(s), s = 0,1,\dots,M-1$, be another set of state metrics on the n-partite trellis at the time i. The computation of the backward β probabilities starts from the initial conditions $\beta_{N-1}(s) = 0$ and following the edges of the trellis with non-zero branch probabilities $\gamma_i(s,s')$, the decoder, at each iteration computes $\beta_i(s)$, according to

$$\beta_i(s') = \max_s\big(\beta_{i+1}(s) + \gamma_i(s,s')\big) \tag{2}$$

where $s' = 0,1,\dots,M-1$. Then the logarithm of likelihood ratio (LLR) for each bit x_k is computed as

$$LLR(x_k) = \max_s\big(\alpha_i(s) + \beta_{i+1}(s) + \gamma_i(s,s')\big) - \max_s\big(\alpha_i(s) + \beta_{i+1}(s) - \gamma_i(s,s')\big) \tag{3}$$

III. THE PARALLEL TURBO DECODER

Fig. 4 shows a block diagram of a turbo decoder. The turbo decoder is coupled to receive channel information of a turbo code. The received turbo code is made of two parallel recursive systematic convolutional codes RSC1 and RSC2. Principle of operation of the turbo decoder is described on fig. 2. The turbo decoder is configured first to decode the first code RSC1, then the code RSC2. The decoder repeats this sequence for predefined number of iterations.

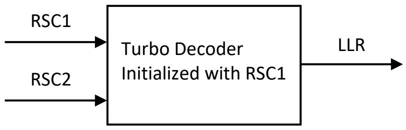


Fig. 4. Block Diagram of a Turbo Decoder initialized with RSC1

Parallel decoding of the turbo code made of two codes RSC1 and RSC2 may be achieved with two serial turbo decoders (as in fig. 4) working in parallel, such that each turbo decoder is initialized with different RSC code (fig. 5).

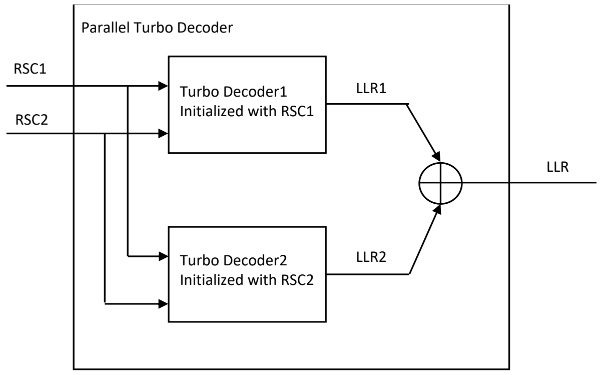


Fig. 5. Block Diagram of a Turbo Decoder initialized with RSC1

The parallel turbo decoder (fig. 5) is coupled to receive channel information for two parallel recursive systematic convolutional codes RSC1 and RSC2. The parallel turbo decoder is made of two serial turbo decoders Turbo Decoder1 and Turbo Decoder2. Each of the serial turbo decoders start operation with different RSC code. The logarithm of likelihood ratio (LLR) for each bit x_k on the output of the parallel decoder is sum of logarithm of likelihood ratio (LLR) for each bit x_k on the output of each serial decoder. In general, the parallel turbo decoder may be generalized for any turbo code made of arbitrary number of recursive systematic convolutional codes in parallel connection. In practice, the parallel turbo decoder (fig. 5) may be implemented with one serial turbo decoder (fig. 2), where, at any moment, one MAP decoder computes the LLR1 coefficients from the Turbo Decoder1 and the other MAP decoder computes LLR2 coefficients from the Turbo Decoder2, simultaneously.

IV. Practical Results

In our simulation of parallel turbo decoder, we use turbo code made of two recursive systematic convolution codes (as in fig. 1). The turbo code is with code rate 1/2, which means that one of the two RSC codes is punctured out at any bit interval. The convolutional encoders are recursive with 16 states. Encoded sequence is 1025 bits long. The Turbo code is sent over Gaussian channel. To store state and branch metric we use IEEE 754 double precision format. Results are shown on fig. 6.

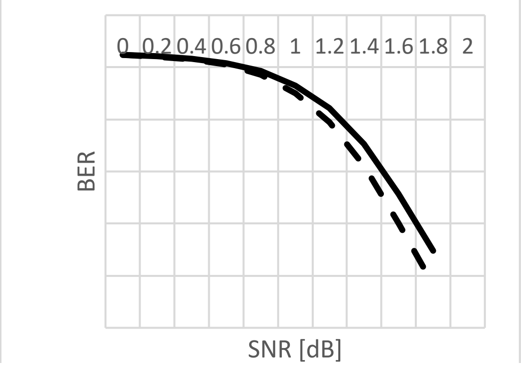


Fig. 6. Performance comparison between regular turbo decoder and parallel turbo decoder.

Fig. 6 compares performance of a regular turbo decoder (fig. 2) and parallel turbo decoder (fig. 5). The parallel turbo decoder (full line) is set to perform 4 iterations, while the regular turbo decoder (intersected line) is set to perform 8 iterations before outputting the logarithm of likelihood ratio (LLR) for each bit x_k. Both the parallel turbo decoder (fig. 5) and the regular turbo decoder (fig. 2) are tested for bit error rate for various signal to noise ratios. Figure 6 shows acceptable 0.1 dB loss between 4 iterations parallel decoder and 8 iterations regular decoder.

V. Conclusion

We have demonstrated a turbo decoding algorithm that improves bit error rate in decoding turbo codes by using parallel turbo decoders (fig. 5). From the perspective of improved decoding speed, it is obvious from fig. 6 that the parallel turbo decoder may double the speed of decoding of the turbo decoders. If we can double the hardware resources, the parallel turbo decoding algorithm may be twice as fast as the classical turbo decoding algorithm.

One line of research is to improve decoding speed of turbo decoders and to achieve the decoding speed of the LDPC codes [10]. Advantage of the parallel turbo decoder (fig. 5) that is introduced in this paper is that it can be applied on any suboptimal turbo decoder. Therefore, the proposed turbo decoder does not require change of already established standards for turbo codes.

References

[1] C. Berrou, A. Glavieux, and P. Thitimajshima, “Near Shannon limit error-correcting coding and decoding: Turbo-codes,” in Proc. ICC, Geneva, Switzerland, May 1993.

[2] A. J. Viterbi, “An Intuitive Justification and a Simplified Implementation of the MAP Decoder for Convolutional Codes,” IEEE Journal on Selected Areas in Communication, vol. 16, pp. 260–264, Feb. 1998.

[3] Forney, G. D., “The Viterbi Algorithm,” Proceedings of the IEEE, Vol. 61, Issue 3, March 1973.

[4] L. R. Bahl, J. Cocke, F. Jelinek, and J. Raviv, “Optimal decoding of linear codes for minimizing symbol error rate,” IEEE Trans. Inform. Theory, vol. 20, no. 3, pp. 284–287, Mar. 1974.

[5] J. Hagenauer, E. Offer, and L. Papke, “Iterative Decoding of Binary Block and Convolutional Codes,” IEEE Trans. Inform. Theory, vol. 42, no. 2, pp. 429–445, Mar. 1996

[6] E. Boutillon, W. J. Gross, and P. G. Gulak, “VLSI Architectures for the MAP Algorithm,” IEEE Trans. On Communications, vol. 51, no. 2, February 2003.

[7] M. Zhan, L. Zhou, “A Memory Reduced Decoding Scheme for Double Binary Convolutional Turbo Code Based on Forward Recalculation,” in 7th International Symposium on Turbo Codes and Iterative Information Processing (ISTC), Gothenburg, Sweden, 2012.

[8] H.-M. Choi, J.-H. Kim, and I.-C. Park, “Low-power hybrid turbo decoding based on reverse calculation,” ISCAS. 2006. pp. 2053–2056.

[9] E. Boutillon, C. Douillard, and G. Montorsi, “Iterative Decoding of Concatenated Convolutional Codes: Implementation Issues,” Proc. of the IEEE, Vol. 95, No. 6, June 2007, pp. 1201-1227.

[10] R. G. Maunder, “A Fully-Parallel Turbo Decoding Algorithm,” IEEE Trans. on Comm., Vol. 63, Issue 8, Aug. 2015, pp. 2762-2775.

Question Answering with Deep Learning: A Survey

Martina Toshevska
Faculty of Computer Science
and Engineering
Ss. Cyril and Methodius University
Skopje, Macedonia
martina.toshevska@finki.ukim.mk

Georgina Mirceva
Faculty of Computer Science
and Engineering
Ss. Cyril and Methodius University
Skopje, Macedonia
georgina.mirceva@finki.ukim.mk

Mile Jovanov
Faculty of Computer Science
and Engineering
Ss. Cyril and Methodius University
Skopje, Macedonia
mile.jovanov@finki.ukim.mk

***Abstract*—Automatically generating answer for a given question is a process in which the computer is supposed to answer a question in a natural language where the question itself is also provided in natural language. Deep learning techniques gained extensive research in both fields of computer vision and natural language processing. Therefore, they are extensively applied for the task of question answering using wide varieties of datasets.**

This survey aims to overview some of the latest algorithms and models proposed in the field, as well as datasets exploited for training and evaluating the models. In this survey, the models are presented as part of one of the following groups: classical deep neural networks, dynamic memory networks and relation networks. Several datasets have been proposed specifically for the research on automatic question answering. This survey briefly overviews datasets for two different categories of question answering: textual and visual. In the end, evaluation metrics utilized in the field are presented, grouped as: metrics for evaluation of an information retrieval system and metrics for evaluating automatically generated text.

***Keywords*—Question Answering, Visual Question Answering, Textual Question Answering, Natural Language Processing, Computer Vision, Deep Learning**

I. Introduction

The ability to provide an answer to a natural language question is known as question answering. Automatically generating answer for a given question is a process in which the computer is supposed to answer a question in a natural language. The question itself is also provided in natural language. Thus, the computer needs to understand the question too. Based on the domain, questions can be classified as questions referring to mathematical or logical tasks, questions asked in online communities (known as Community Question Answering), questions in the medical domain etc. In accordance with the type of information they refer to, questions can be separated into questions concerning text (known as Textual Question Answering), questions concerning images (known as Visual Question Answering), etc.

Textual Question Answering (TQA) is the task of extracting a text snippet from a passage which corresponds to a specific question. This task differs from classical information retrieval since the output is a particular piece of information rather than a collection of documents. Its purpose is to create textual answer for a specific question. The question is either from a specific domain (such as science, math, etc.) or from a general domain. With the advent on online communities, such as Quora[1], Stack Overflow[2] and Stack Exchange[3], a new type of textual question answering has increased popularity - Community Question Answering (CQA). The goal of this task is to resemble actions performed by users in the community such as ranking answers according to their relevance, selecting the best answer for a specific question, identifying duplicate questions etc.

Visual Question Answering (VQA) is the task where questions are asked about given image. It has received attention from researches in both natural language processing and computer vision communities. In the most common form of this task, the computer is given an image and a question about the image. It is supposed to create an answer for the question, which is typically a word or phrase. Images are either natural or synthetic i.e. computer generated. The latter are referred as abstract scenes. The idea behind creating such synthetic sets is to focus only on high-level reasoning rather than low-level image processing.

Deep learning techniques [1] gained extensive research in both fields of computer vision and natural language processing. Therefore, they are extensively applied for the task of question answering using wide varieties of datasets. This survey aims to overview some of the latest algorithms and models proposed in the field, as well as datasets exploited for training and evaluating the models.

The rest of this survey is organized as follows. Section 2 gives an overview of algorithms applied in the field of question answering. Section 3 explores some of the datasets being used. Evaluation metrics are presented in Section 4 and at the end, Section 5 concludes the survey.

II. Algorithms

Deep learning techniques gained extensive research in the domain of question answering, either visual or textual. The following subsections present some of the models utilized in the field. The models are grouped into three groups based on the architecture: deep neural networks, dynamic memory networks and relation networks.

[1] https://www.quora.com/, last visited: 24.01.2019
[2] https://stackoverflow.com/, last visited: 24.01.2019
[3] https://stackexchange.com/, last visited: 24.01.2019

A. Deep Neural Networks

There is a variety of deep neural models proposed in the field of question answering. The most common architectures consist of Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs). The first two models described in this subsection address the problem of Community Question Answering (CQA), while the third model focuses on Visual Question Answering (VQA).

The first model, SwissAlps [2], utilizes CNN for computing similarity between question and its candidate answer. First, both the question and the answer sequences are processed by an embedding layer creating matrix representation for the sentences. Attention matrix is then calculated by computing pairwise similarity between the word embeddings of these matrices based on Euclidean distance. The attention matrix is multiplied by two weight matrices in order to generate attention features. These features are stacked on top of the sentence matrix creating three-dimensional array. The purpose is to give higher weight to the relevant part of the sentences. Such arrays are fed into a convolutional layer which creates a feature vector for the sentences by applying a set of convolutional filters. Attention values are generated by summing the attention matrix column-wise for the question and row-wise for the answer candidate. These values are used to weight the feature map matrices obtained by the convolutional layer. Standard max pooling is applied to the attention weighted feature map matrices. Finally, the map matrices are fed through a fully connected layer followed by a softmax regression layer.

The second model, FuRongWang [3], allows using either CNN or RNN for the processing of question and answer. In the same way as the previously described model, both question and answer are represented with a matrix composed of word embedding vectors. An augmented feature vector is added at the tail of these matrices. For each word in the question, the question augmented feature vector has value 1 at the corresponding position if it is present in the answer and value 0 otherwise. The same holds for the answer augmented feature vector. Next part of the model is a neural network which can be convolutional or recurrent. The convolutional network is represented as a convolutional layer consisted of several feature maps followed by max pooling. The recurrent network is represented as a bidirectional LSTM which processes the sentence in both directions. The output is a vector representation of the sentences. Another part is an interaction layer which calculates the relevance between question and answer by multiplying a weight matrix with the answer feature vector from right and with the transpose of the question feature vector from left. Features extracted by the neural networks altogether with extra features from the interaction layer and augmented features are concatenated and fed into a fully connected layer. In the end, softmax function is applied.

The SAN model [4] applies multi-step attention for the problem of VQA. A convolutional neural network produces an image feature map for the image regions and another convolutional network or LSTM encodes the question. The answer is then generated with an attention mechanism as described below. The image feature map and the question vector are combined, and then attention weights are produced with a softmax function. A weighted sum of region vectors is calculated based on the attention weights. This sum is then combined with the question vector forming a vector called refined query vector, which encodes information about the question and the relevant part of the image. For complicated questions that require more complex reasoning, single attention is not sufficient. Therefore, previously described attention mechanism is repeated multiple times before inferring the final answer by a softmax function.

B. Dynamic Memory Networks

Dynamic Memory Network (DMN) [5] is a framework based on neural networks capable of solving sequence tagging tasks, classification problems, sequence-to-sequence tasks and question answering tasks that require transitive reasoning. The DMN first computes a representation for all inputs and the question. An input module encodes input sequences into distributed vector representations. The raw text input is first transformed into word embedding vectors and is then fed through recurrent neural network that creates vector representation. End-of-sentence token is inserted after each sentence. The final vector representation is composed of the hidden state at each end-of-sequence token. A question module encodes the question into a distributed vector representation. Analogous as in the input module, the question is first converted into word embedding vectors and then fed into a recurrent network for creating the representation. Gated Recurrent Unit (GRU) is used in both input and question module as a recurrent neural network. The question representation then triggers an iterative attention process that searches the inputs and retrieves relevant facts. Given a collection of input representations, an episodic memory module chooses which parts of the inputs to focus on through the attention mechanism. It is a two-layer feed forward network that computes scalar score based on the input vector, previous memory and question vector. This score is used to weight the input sequence using a GRU in order to compute the episode. The episode memory module may pass over the input multiple times, updating episode memory after each pass. Each iteration provides the module with newly relevant information about the input and by the final iteration the episodic memory should contain all the information required to answer the question. Finally, an answer module generates the answer based on the final memory vector of the episodic memory module and the question itself.

DMN+ [6] is a modification of DMN that proposes modification of input representation, attention mechanism and memory update. The first modification is replacing the GRU in the input module with two different components: sentence reader (positional encoder adapted from [7]) and input fusion layer (bi-directional GRU). Beyond text, DMN+ can process image as input. This visual input module is composed of three parts: local region feature extraction (extracting features with convolutional neural network based on VGG-19 [8]), visual feature embedding (linear layer with tanh activation that projects the local regional vectors to the textual feature space used by the question vector) and input fusion layer (bi-directional GRU). The second modification is updating the memory in the episodic memory module with Rectified Linear Unit (ReLU) layer instead of GRU. The last modification is the attention mechanism. The attention is implemented by associating a single scalar value called attention gate with each input fact. Two different mechanisms are proposed: soft attention, which produces a contextual vector through a weighted summation of the sorted list of input fact vectors and corresponding attention gates, and attention based GRU that is a modification of the standard GRU by incorporating attention gates.

C. *Relation Networks*

Relation Network (RN) [9] is a neural network module with a structure primed for relational reasoning. The main idea behind this network is the ability to compute relations without the need to be learned in a way in which recurrent neural networks learn to capture sequential dependencies and convolutional neural networks learn spatial dependencies. One model encompassing RNs is presented in [10]. In the first step, the image is embedded using a Faster R-CNN embedding method [11] creating a feature map for each region of interest, while the question is embedded using a GRU. Next step is applying visual attention to focus on important image regions. The attention mechanism takes as input the question embedding and embedded visual regions, and then weights the visual regions according to their relevance. An RN module performs pair-wise reasoning on objects. With the use of previously computed attention weights the most relevant regions of interest are selected. Then, for each pair of region embeddings, relational embedding is computed based on the region embeddings and the question. In the end, final relational embedding is produced by summing the embeddings for each pair. A joint embedding is computed with multimodal fusion by combining the question embedding, the attended image embedding and the relational embedding. These are combined using the Hadamard product. The final part of the model is a classifier that performs multi-label classification to infer the answer for the question according to the joint embedding.

III. Datasets

Several datasets have been proposed specifically for the research on automatic question answering. The following subsections briefly overview datasets for two different categories of question answering: textual and visual. Datasets for textual question answering typically comprise a set of questions with at least one answer, while datasets for visual question answering consist of a set of images, questions about them and their corresponding answers. Sometimes, datasets include additional information such as text articles, scene graph annotations, objects, object attributes, object relations etc.

A. *Textual Question Answering*

SemEval (International Workshop on Semantic Evaluation) is an ongoing series of evaluations of computational semantic analysis systems. SemEval-2017 is the eleventh workshop in the series. It comprises different tasks for semantic evaluation including Community Question Answering [12]. This task is divided into five subtasks each providing different data for ranking questions, question comments, question external comments, correct answers, as well as identifying duplicate questions.

SQuAD (Stanford Question Answering Dataset) [13] is a dataset consisted of question-answer pairs posed by crowd workers on Wikipedia articles. The dataset does not provide a list of answer choices for the question. On the contrary, the answer must be selected from all possible spans in the passage. It can be a segment of text, or span, from the corresponding reading passage. The second version of the dataset [14] introduces unanswerable questions. That is, the first version is extended with additional unanswerable questions. An additional challenge, besides determining the answer, when working with this dataset is determining when the answer is not available and abstain from answering.

QuAC (Question Answering in Context) [15] is a dataset comprising QA dialogues between two individuals. The first individual asks free-form questions, while the second individual gives an answer for the question. The goal is to predict text span which answers a question about Wikipedia article. The question can also be unanswerable.

The bAbI project is organized towards the goal of automatic text understanding and reasoning. It comprises different tasks where each task is associated with a specific dataset. The Simple Questions dataset [16] refers to open-domain question answering and is based on the Freebase knowledge database. It consists of a total of 108,442 questions written in natural language by human English-speaking annotators. The answer for each question is a fact formatted as tuple (subject, relationship, object) that also provides a complete explanation.

B. *Visual Question Answering*

CLEVR (Compositional Language and Elementary Visual Reasoning) [17] provides a dataset that requires solving complex reasoning problems such as attribute identification, counting, comparison, spatial relationships, and logical operations. The dataset contains synthetic images generated by randomly sampling and rendering a scene graph. Scene graph is a kind of image scene representation in the form of a graph where nodes represents object and edges connect objects that are spatially related. CLEVR contains three object shapes: cube, sphere, and cylinder. They can come in two absolute sizes (i.e. small and large), two materials (i.e. shiny metal and matte rubber) and eight colors. The objects are spatially related via four relationships: left, right, behind and in front. Each question is associated with a functional program that can be executed on an image's scene graph, yielding the answer to the question.

VQA [18] is the most widely used dataset for the task of VQA. It is divided into two datasets according to the nature of images: natural or abstract. The set of natural images (VQA-real) contains images from the MS COCO [19] dataset, while the set of abstract images (VQA-abstract) contains images with abstract scenes. The reason for creating the abstract scenes dataset is to avoid the low-level vision tasks and focus only on high-level reasoning. Each image has questions with both ground truth and plausible but likely incorrect answers. The questions are provided from human annotators. Answers are typically a word or a short phrase. However, for many of the questions, a yes or no answer is sufficient.

Visual Genome [20] is a dataset containing real-world images obtained as intersection of images in MS COCO [19] and YFCC100M [21]. It aims to connect structured image concepts to language. For each image, the dataset provides region descriptions, object instances, attributes, relations, region graphs, scene graphs and visual question answers. There are two types QA pairs associated with each image: based on the entire image (i.e. free-form) and based on specific image regions (i.e. region-based). Each image has at least one question of each type: what, where, how, when, who and why.

IV. Evaluation Metrics

Evaluating the quality of question answering systems is an important aspect of the problem. Their performance is measured with different types of evaluation metrics. According to their nature, the answers can be split into two groups: short answers composed of only one word and long answers composed of multiple words. Based on this division, we split the metrics into two groups: metrics for evaluation of an information retrieval system and metrics for evaluating automatically generated text. Creating answers from the first group, short answers, comes down to classification. Such answers are evaluated with classification and information retrieval metrics. Answers containing multiple words are evaluated with evaluation metrics for automatically generated text. Several evaluation metrics from both groups are described in the following subsections.

A. Metrics Based on Information Retrieval

The most common way for evaluating a machine learning model is to use metrics based on information retrieval. These metrics are applied when the purpose of the system is ranking answers, ranking similar questions or answers, or when the answer generation comes down to classification. Example evaluation metrics include accuracy, precision, recall, F1-measure, etc [22].

However, simple precision and recall do not apply for systems that rank the retrieved documents. That is, if we are comparing the performance of two ranked retrieval systems, we require a metric that will prefer the one that ranks the relevant documents higher [23]. One such metric is MAP (Mean Average Precision). It is calculated as follows. First, we descend through the ranked list of items and note the precision only at those points where a relevant document has been encountered. For a single query, these individual precision measurements are averaged over the return set up to some fixed cutoff. The final measure is the mean of such averages. Another metric assuming that the system retrieves relevant documents is MRR (Mean Reciprocal Rank). Each query is scored according to the reciprocal of the rank of the first correctly retrieved document. The final measure is the mean of such reciprocal ranks.

B. Metrics Based on Natural Language Generation

The evaluation of computer-generated natural language sentences is an inherently complex task. The most common way to assess the quality of automatically generated texts is the subjective evaluation by human experts. However, human evaluation is not always attainable. Another approach is to use automatic evaluation metrics. These metrics compute a score that indicates the similarity between generated and reference text. They are applied when the purpose of the system is to generate natural language phrase.

The METEOR [24] automatic evaluation metric is designed for evaluating machine translation. It is based on the harmonic mean of unigram precision and recall, where recall is weighted higher. It scores generated translations by aligning them to one or more reference translations. Alignments are based on exact, stem, synonym, and paraphrase match between words and phrases. BLEU [25] was also designed for automatic evaluation of machine translation. It measures how close a candidate sequence is to a reference sequence, i.e. the hits of n-grams of a candidate sequence to the reference. BLEU can be calculated with different length of n-grams. BLEU-N is the score where N is the maximum length of considered n-grams.

ROUGE-L [26] is a recall-oriented metric developed for evaluation of text summarization. It applies the concept of Longest Common Subsequence (LCS). The intuition is that the longer the LCS between two summary sentences is, the more similar they are. The score is 1 when the two sequences are equal, and 0 when there is nothing in common between them.

CIDEr [27] was developed specifically for evaluation of image descriptions. The goal is to automatically evaluate how well a candidate sentence matches the consensus of a set of image descriptions, i.e. how often n-grams in the candidate sentence are present in the reference sentences. All words in the sentences (both candidate and references) are first mapped to their stem or root forms. SPICE [28] is another metric developed for evaluation of image captions. It measures how effectively image captions recover objects, attributes and the relations between them. It is based on the agreement of the scene-graph tuples of the candidate sentence and all reference sentences. Scene-graph is a semantic representation that parses the given sentence to semantic tokens. A set of tuples is formed by using the elements of the graph and their possible combinations. The score is defined as the F1-score based on the agreement between the candidate and reference caption tuples.

V. Conclusion

Automatically generating answer for a given question is a process in which the computer is supposed to answer a question in a natural language where the question itself is also provided in natural language. In accordance with the type of information the questions refer to, the task can be classified as Textual Question Answering, Visual Question Answering, etc. Textual Question Answering is the task of extracting a text snippet from a passage, which corresponds to a specific question. A specific type of textual question answering is Community Question Answering that refers to online communities. Visual Question Answering is the task about questions concerning images, either natural or abstract.

Deep learning techniques gained extensive research in the domain of question answering. There is a variety of deep neural models proposed in this domain. In this survey, the models are presented as part of one of the following groups: classical deep neural networks, dynamic memory networks and relation networks. Representative models of each group are considered.

Several datasets have been proposed specifically for the research on automatic question answering. Datasets for textual question answering are provided by SemEval, SQuAD, bAbI, etc. They typically comprise a set of questions with at least one answer. CLEVR, VQA, Visual Genome and others provide datasets for visual question answering. These datasets consist of a set of images, questions about them and their corresponding answers.

In the end, evaluation metrics utilized in the field are presented. They come as a part of one of the following groups: metrics for evaluation of an information retrieval system (such as accuracy, precision, recall, F1-measure, mean average precision and mean reciprocal rank) and metrics for evaluating automatically generated text (such as METEOR, BLEU, ROUGE-L, CIDEr and SPICE).

References

[1] J. Patterson and A. Gibson, *Deep Learning: A Practitioner's Approach*. O'Reilly Media, Inc., 2017.

[2] J. M. Deriu and M. Cieliebak, "Swissalps at semeval-2017 task 3: Attention-based convolutional neural network for community question answering," in *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pp. 334–338, 2017.

[3] S. Zhang, J. Cheng, H. Wang, X. Zhang, P. Li, and Z. Ding, "Furongwang at semeval-2017 task 3: Deep neural networks for selecting relevant answers in community question answering," in *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pp. 320–325, 2017.

[4] Z. Yang, X. He, J. Gao, L. Deng, and A. Smola, "Stacked attention networks for image question answering," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 21–29, 2016.

[5] A. Kumar, O. Irsoy, P. Ondruska, M. Iyyer, J. Bradbury, I. Gulrajani, V. Zhong, R. Paulus, and R. Socher, "Ask me anything: Dynamic memory networks for natural language processing," in *International Conference on Machine Learning*, pp. 1378–1387, 2016.

[6] C. Xiong, S. Merity, and R. Socher, "Dynamic memory networks for visual and textual question answering," in *International conference on machine learning*, pp. 2397–2406, 2016.

[7] S. Sukhbaatar, J. Weston, R. Fergus, *et al.*, "End-to-end memory networks," in *Advances in neural information processing systems*, pp. 2440–2448, 2015.

[8] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014.

[9] A. Santoro, D. Raposo, D. G. Barrett, M. Malinowski, R. Pascanu, P. Battaglia, and T. Lillicrap, "A simple neural network module for relational reasoning," in *Advances in neural information processing systems*, pp. 4967–4976, 2017.

[10] L. J. Petersen, "Attended Relational Reasoning for Visual Question Answering," Master's thesis, Aalborg University, 2018.

[11] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in neural information processing systems*, pp. 91–99, 2015.

[12] P. Nakov, D. Hoogeveen, L. Màrquez, A. Moschitti, H. Mubarak, T. Baldwin, and K. Verspoor, "Semeval-2017 task 3: Community question answering," in *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pp. 27–48, 2017.

[13] P. Rajpurkar, J. Zhang, K. Lopyrev, and P. Liang, "Squad: 100,000+ questions for machine comprehension of text," in *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pp. 2383–2392, 2016.

[14] P. Rajpurkar, R. Jia, and P. Liang, "Know what you dont know: Unanswerable questions for squad," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, vol. 2, pp. 784–789, 2018.

[15] E. Choi, H. He, M. Iyyer, M. Yatskar, W.-t. Yih, Y. Choi, P. Liang, and L. Zettlemoyer, "Quac: Question answering in context," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 2174–2184, 2018.

[16] A. Bordes, N. Usunier, S. Chopra, and J. Weston, "Large-scale simple question answering with memory networks," 2015.

[17] J. Johnson, B. Hariharan, L. van der Maaten, L. Fei-Fei, C. Lawrence Zitnick, and R. Girshick, "Clevr: A diagnostic dataset for compositional language and elementary visual reasoning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2901–2910, 2017.

[18] S. Antol, A. Agrawal, J. Lu, M. Mitchell, D. Batra, C. Lawrence Zitnick, and D. Parikh, "Vqa: Visual question answering," in *Proceedings of the IEEE international conference on computer vision*, pp. 2425–2433, 2015.

[19] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *European conference on computer vision*, pp. 740–755, Springer, 2014.

[20] R. Krishna, Y. Zhu, O. Groth, J. Johnson, K. Hata, J. Kravitz, S. Chen, Y. Kalantidis, L.-J. Li, D. A. Shamma, *et al.*, "Visual genome: Connecting language and vision using crowdsourced dense image annotations," *International Journal of Computer Vision*, vol. 123, no. 1, pp. 32–73, 2017.

[21] B. Thomee, D. A. Shamma, G. Friedland, B. Elizalde, K. Ni, D. Poland, D. Borth, and L.-J. Li, "The new data and new challenges in multimedia research," 2015.

[22] J. Han, M. Kamber, and J. Pei, "Data mining concepts and techniques third edition," *Waltham: Elsevier*, 2012.

[23] J. H. Martin and D. Jurafsky, *Speech and language processing: An introduction to natural language processing, computational linguistics, and speech recognition*. Pearson/Prentice Hall Upper Saddle River, 2009.

[24] M. Denkowski and A. Lavie, "Meteor universal: Language specific translation evaluation for any target language," in *Proceedings of the ninth workshop on statistical machine translation*, pp. 376–380, 2014.

[25] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "Bleu: a method for automatic evaluation of machine translation," in *Proceedings of the 40th annual meeting on association for computational linguistics*, pp. 311–318, Association for Computational Linguistics, 2002.

[26] C.-Y. Lin, "Rouge: A package for automatic evaluation of summaries," *Text Summarization Branches Out*, 2004.

[27] R. Vedantam, C. Lawrence Zitnick, and D. Parikh, "Cider: Consensus-based image description evaluation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4566–4575, 2015.

[28] P. Anderson, B. Fernando, M. Johnson, and S. Gould, "Spice: Semantic propositional image caption evaluation," in *European Conference on Computer Vision*, pp. 382–398, Springer, 2016.

Shopping Of Tomorrow – Internet Of Things Approach

Veno Pachovski
School of Computer Science and Information Technology
University American College Skopje
Skopje, Macedonia
pachovski@uacs.edu.mk

Irena Stojmenovska
School of Computer Science and Information Technology
University American College Skopje
Skopje, Macedonia
irena.stojmenovska@uacs.edu.mk

Abstract*— Shopping is a one of the most crucial activities of everyday living in modern societies. We spend a lot of time preparing, browsing between the shelves looking for some particular product (based on some recommendation, price or brand), calculating approximate cost of the goods, then waiting in line at the cashiers to pay, finally taking them out of the cart, for registering them by their barcodes and then, putting them to the cart again, finally paying (usually) by a credit card. What if there are intelligent trolleys – shopping carts which could be used to connect customer using IoT technology with the Information system of the market? Starting by optimized path for all the goods in the shopping list, suggesting alternatives by brand or price, while automatically calculating the total cost – the possibilities for optimization are numerous. Finally, implementing m-payment procedures, there will be no waiting at all at the exit. Such a model is proposed in this paper. (Abstract***)**

***Keywords*— *architecture, IoT, business model, shopping, optimization* (*key words*)**

I. Introduction

Shopping as a regular activity takes valuable time. Not only at the market, but also at home preparing, analyzing what one needs, and hopefully, preparing a list. Then, at the market, browsing through shelves, looking for the listed items, brands, affordable prices, … Finally, at the end of the shopping there is a boring activity – taking all of the items out the shopping cart, placing them at the counter so that the cashier can pass them through some form of identification process (item, quantity, price) thus entering them into the system, so that the bill could be generated and a customer charged. Chances are that sometimes something which is necessary will not be bought (simply forgotten), or something will be bought because of some other reasons. The progress of technology enables that the stress can be minimized, and the (possible) pleasure while shopping even increased.

Welcome to the consumer society of 21-st century.

II. Internet of Things

A. IoT – brief overview

What is Internet of things? The Internet of Things (IoT) is the network of physical objects that contain embedded technology to communicate and sense or interact with their internal states or the external environment [1]. In this case, the meaning of the concept is broadened into a full architecture, which includes the traditional concept enriched with additional hardware and business model.

2018 ranking of Top IoT Segments (shown on the graph further on), was based on continuous track of the IoT ecosystem, by mining thousands of homepages. It managed to assemble, verify, and classify 1,600 actual enterprise IoT projects [2].

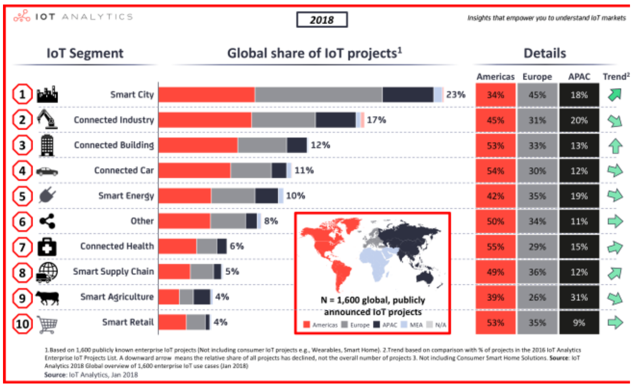


Fig. 1. Top 10 IoT Segments in 2018 [2]

Most of those IoT projects were identified in Smart City category (367 projects), followed by Connected industry (265) and Connected Building IoT projects (193). In comparison to the 2016 ranking (by the same source), Smart City (driven by government and municipality-led initiatives) has surpassed Connected Industry as the number one IoT segment of identified projects while Connected Building (driven by widespread uptake of building automation solutions that increase operational efficiency and reduce costs) has climbed four places to become the third biggest IoT segment [2].

So, it is obvious that there is progress towards solving everyday life problems. In that context, creating solutions for smart shopping seems to be next logical step.

B. Solution architecture

The architecture of the solution presented here directly stems from the IoT, and in a sense, enriches it.

First, there is software solution, which includes smart phone application, m-phone payment implemented and information system. Then, there is hardware – the smart shopping cart itself, which will also serve as a platform for communication between the customer and the Market. Finally, there is an exit sector, equipped with RFID readers, which needs a software solution of its own.

III. Smart Shopping Cart

The key element of this model is a smart shopping cart. It is a classic shopping cart, equipped with a slot for placing (or connecting) a smart phone, (depending on a strategy) one (or even better), two small intelligent screens – tablets would do the job nicely - placed strategically, one of them at the customers' side (the back-end, where the push handles are), the other at the front end.

The first screen (smart device) is a principal carrier of information, and should be used for all the communication between customer and the Market. That includes implementing some kind of guidance toward a product, or even defining a shortest (optimal) path for collecting all products. The second one could be used for presenting additional information, or if there is a child accompanying the parent – to be amused (while shopping) by watching (educational or fun) content, appropriate for its age.

Additionally, an RFID tag reader should be installed, so that when a product is placed in the cart, it is automatically identified and shown on the screen, updating the current value of the goods in the cart.

Before continuing with explanation of the approach, existing solutions similar to this will be listed. Namely, there are models proposed (or even already active), although none of them quite like the one proposed here.

A. Some existing solutions

The idea that applying artificial intelligence to brick & mortar in order to ease the whole shopping experience also has spawned another breed of start-up families.

Namely, starting from the Amazon Go proposed concept, there are several start-ups, eyeing the traditional supermarket business and claim to transform the store into something akin to it, albeit less ideal, yet in a much cost-effective way. So, their accent is mostly on improving the traditional shopping cart.

One of them (inspired by Amazon Go) is a solution proposed by a startup called Caper. Caper has developed an intelligent shopping cart that lets customers scan items as they're shopping and pay for whatever they've picked up before they leave. The cart features an interactive display and a card reader [3].

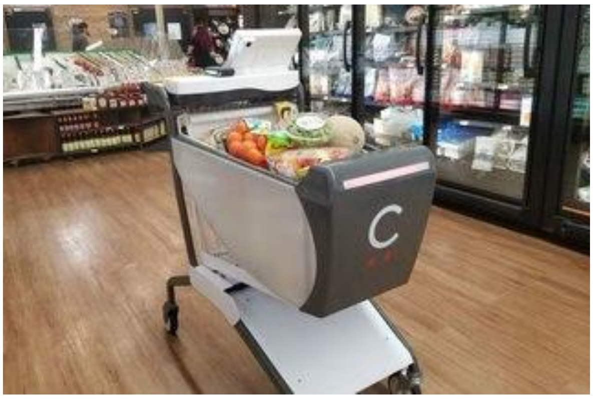


Fig. 2. Model of the smart shopping cart, by Caper [3]

The cart also shows customers deals on similar products to what they're scanning or products nearby. Caper claims that this leads to customers purchasing 18 per cent more products when using the automated carts [4]. The start-up is currently training its deep learning algorithm to identify items with the help of image recognition cameras and a weight sensor, in a bid to go completely scanless [4].

There is also Focal Systems from Silicon Valley. Established in 2015, it deploys the latest computer vision and machine learning technology via an easy-to-use tablet mounted on shopping cart handle-bars. It has 4 key functions:

1) *Real-time out of stock detection*
2) *Location based promotion and advertising*
3) *In-store navigation and product search*
4) *Automated checkout*

Fig. 4. Focal systems solution, no smart-phones [5]

This solution does not require implementing any changes to the store. Their approach is to supply appropriate software and automate the process on the customer side.

Following closely to this trend, there is a Chinese company startup Chaohi.

Fig. 3. Chaohi is one of Chinese startups [5]

It launched its product in February 2017. The smart shopping cart asks the customers to use Wechat to log in the tablet. Then customers simply scan the products to do the self check-out. Similar to Focal Systems, it can offer product recommendation, show discounts, theft prevention; except the real-time monitoring of shelf display for the retailers and in-store navigation & product search for the customers. According to Chaohi, ERP integration and installation process for the retailers are also very easy, which usually can be finished within 2 weeks [5].

These shopping carts come pretty close to the model proposed here, but none of the architectures they are implemented in comes near the one discussed here.

B. Information system on the side of the Market

The market usually has an Information system in place, and the customer should be able to connect. Depending on the goal of the market, that information system could have more elements then those needed for the simple shopping process.

The basic information needs will be covered by default (they already exist in a database) – like goods, prices, declarations, descriptions – usual info about goods, which could be enriched by a video or some other details that could be used to persuade the customer to decide on the side of buying the product.

Also, there should be some more data which could be used to help the customer navigate easier. For example, some kind of spatial related information which could be used to guide the customer directly to the shelf that situates the required good – that would definitely shorten the time the customer spends in the market.

And, of course, an implementation of AI for analyzing shopping habits of the customer (and customers, in general), and making suggestions.

C. *Apllication support*

Market should supply a specialized application for the smart devices, so that the connection with its IS should be seamless and established either at the moment the customer enters or, at latest, the moment the customer puts the smart device in a slot on the shopping cart. Also, depending on the port, if the smart cart has a battery, this moment can be used to ask the customer whether to re-charge the smart device (or not). Accordingly, that kind of service could be offered free or charged, as well.

Then, the application transfers control to the smart device of the cart and all further communication will be conducted through it.

Finally, when the customer decides that it is time to leave (loaded all the goods in the cart, or simply because it is time), that application will be used to establish condition for generating the bill, and, ultimately, for paying. And, depending on the strategy of the market, that application could be used to store data about points scored or coupons awarded (if the market has that kind of customer encouragement).

D. *Exit zone of the market*

The exit zone is a crucial part of this model. Namely, the main idea is not to have to unload the cart when checking out.

That is why all products should be equipped with RFID tags (some self-adhesive combinations of RFID-barcode stickers already exist and are very cheap), so that, when a customer decides to leave, simply goes to the exit zone, and activates a procedure for generating a bill.

At that moment, all RFID tags could be scanned again, bill is generated and the application is activating m-payment option. When the customer confirms payment, if everything is in order, then the customer is allowed to leave the exit zone, and RFID tags are marked as used, so that when a customer enters the market next time, if some of the products are on him/her, they will not be charged the second time.

IV. Business aspects – improving marketing strategies

There is an example of positioning the beer near the diapers – counting that when a man goes to the market for diapers (mothers of infants usually stay at home – fathers go shopping for diapers), chances are that, if the beer is near, they will also get some of it (six-pack) [6]. Although, all the references point out that there is no such proven correlation (the result appeared by comparing numbers of items bought calculated by running SQL queries on the databases), actually it became a kind of self-fulfilling prophecy. So, similar research could be applied or marketing techniques could be used when guiding (or following the progress of) the customer through the market.

A. *Benefits for the Market*

First of all, using this approach, the Market increases the amount of information about the shopping habits, needs of its customers, flow of goods – even, increases brand loyalty (in a sense of a market, not some product). Also, following the needs of the customer, the market can suggest similar goods, or similar brands, or even something which is not similar at all, but some kind of AI thinks that could be useful (or interesting) for a customer. On the long run, it will definitely benefit the market.

Also, analyzing the shopping list of customer (assuming that there is such, prepared in advance) when uploaded to the smart shopping cart, the system can compare the goods in the list with those in the market, and prepare a possible list of substitutes, if some of those the customer required are already sold out (or maybe obsolete - not produced anymore). Finally, some products could be offered as substitutes, because they are of the same quality (or same country of origin), but cheaper.

One of the biggest (direct) benefits for the Market itself is saving – time, personal, maybe even space on the shelves…

- Saving time because the customers will be served faster and the service will be better.
- Saving on personal because less people will be needed, whether inside or at the cashiers (check-out) desks – which could still exist, but much less in number.
- Saving space on shelves because in that situation the visibility of the products is not so much an issue if you have some kind of guidance system implemented in the smart shopping cart so that the client is led directly to the required item.
- Saving on the promotional materials – there is no need for printing any catalogues – a customer can download automatically the latest version and can browse through or be offered goods, based on previous visits to the Market.

Finally, the possibility for placing adds or offering products are limitless or at least completely new – creating a tailor made add, suitable for the individual clients.

B. *Benefits for the customer*

First of all, the customer gets a faster and better service – guidance through the market, guarantee that there will be no forgotten item, which usually happens. Also, there is saving on time, no need for physical effort on putting items in the cart, taking them out, then again loading them in the cart for transport to the car. And, no waiting in line, or at least, maximized speed at check-out (in this case, exit zones).

Comparing the prices (or goods) offered by the market AI, could also result in better service altogether.

There could even be a version of the cart which will have some degree of autonomy – for example, needs no pushing, but follows a disabled (or elderly) person while shopping.

Also, it could be equipped with some kind of mechanical hand to reach for products and place them in the cart.

Conclusion

Obviously, smart shopping carts are the future. There is benefit for all parties involved. The progress of big data analytical tools, as well as AI, guarantees that the software aspects of this solution will, become better and better in time. The concept offered in this paper, in general, is completely covered and feasible, although as such, does not exist yet. It remains to be seen when this will become reality.

References

[1] Gartner, Inc. (2019) Internet of Things, https://www.gartner.com/it-glossary/internet-of-things/, Retrieved on April 14, 2019.

[2] Scully, P. (2018) The Top 10 IoT Segments in 2018 – based on 1,600 real IoT projects, https://iot-analytics.com/top-10-iot-segments-2018-real-iot-projects/ , Retrieved on April, 14, 2019.

[3] Tillman, M. (2019) This smart shopping cart eliminates the need for cashiers in store, https://www.pocket-lint.com/gadgets/news/146742-this-smart-shopping-cart-eliminates-the-need-for-cashiers-in-stores, Retrieved on April, 6, 2019.

[4] Shah, S. (2019) Caper's smart shopping cart uses AI to skip checkout lines https://www.engadget.com/2019/01/11/caper-smart-shopping-cart/, Retrieved on April, 14, 2019.

[5] Wu, C. (2017) Smart Shopping Cart, less ideal than Amazon Go, but still a growing obsession, https://www.sanpeiventures.com/smart-shopping-cart-less-ideal-amazon-go-still-growing-obsession/, Retrieved on April, 14, 2019.

[6] Swayer, S. (2016) Beer and Diapers: The Impossible Correlation, https://tdwi.org/articles/2016/11/15/beer-and-diapers-impossible-correlation.aspx , Retrieved on April, 14, 2019.

Synchronization of coupled neural networks

Petar Jovanovski [1], Biljana Tojtovska [2]
1 Macedonian Academy of Sciences and Arts
Email: pjovanovski@manu.edu.mk
2 Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje
Email: biljana.tojtovska@finki.ukim.mk

***Abstract*—In this paper we discuss the problem of synchronization of neural networks and present it on the Izhikevich model of spiky neurons and define it on the Cohen-Grossberg model of coupled neural network. We simulate the models, discuss which parameteres influence the synchronization and refer to different results from the literature.**

***Index Terms*—Synchronization, noisy neural networks, stochastic neural networks, Izhikevich model, Cohen-Grossberg model**

I. Introduction

One aspect of coupled dynamical systems which received great attention, especially in recent decades, is synchronization. This phenomena occurs very often in nature and has many applications in biology, physics and engineering. Of special interest are the models of pulse-coupled biological oscillators where the oscillators communicate by sudden impulses, as in the case of neurons. An example is the Peskin's model of cardiac pacemaker cells - under the assumption that all the nodes (oscillators) are identical, it was shown in [18] that the cells mutually synchronize and fire together no matter from which state they have started. This model was later used for study of a model of coupled neurons [1, 5], which also included synaptic delays, local coupling, refractory periods etc.

Neural oscillations are observed throughout the central nervous system at all levels, and include spike trains, local field potentials and large-scale oscillations which can be measured by electroencephalography (EEG) [6]. Neural oscillations and synchronization have been linked to many cognitive functions such as information transfer, perception, motor control and memory [2, 12, 17]. Artificial neural network models can exhibit chaotic behavior, which may depend on the noise, time delays, impulses or the coupling structure in the case of complex networks. Stability of the processes and synchronization between neurons is essential for information processing, both in biological and artificial neural networks.

The goal of this paper is to present the problem of synchronization on the models of spiky neurons with an emphasis on the Izhikevich model, and to introduce the problem of synchronization of artificial neural networks with an emphasis on the Cohen-Grossberg model. We inspect the synchronization of spikes in a neural network composed of inhibitory neurons [23, 32] with the measure proposed in [19]. We use the Izhikevich neuron [15] as a phenomenological neuron model because of its ability to reproduce the dynamic behavior of many types of biological neurons, all the while being computationally efficient. We then couple these neurons in a random graph and model the synapses via exponential decay [26]. Additionally, recordings of neuronal activity are characterized by a high degree of irregularity – neurons receive tens of thousands of highly fluctuating inputs. The spikes of individual neurons is far from being periodic and relations between the firing patterns of several neurons seem to be random [3, 13, 16, 30]. There is a plethora of models [4, 11, 20, 28] that account for this noise by assuming that the noise comes from random spike thresholds, parameter noise, stochastic spike arrivals, diffusive noise or stochastic resonance. Going forwards we assume that the membrane voltage is governed by diffusive noise, giving rise to a stochastic differential equation [29].

In the case of artificial neural networks we consider the Cohen-Grossberg model of coupled neural network. The analysis is based on the theory of coupled dynamical systems [21, 22]. We define the problem of synchronization, both for the deterministic and stochastic case and refer to some important results from the literature.

The organization of the paper is as follows. Section II gives details on the spiky neural networks through the model of Izhikevich, represented by differential equations. Both the deterministic and the stochastic case are considered, and we also include the case of coupled neurons in a random network. We introduce a measure of synchronizations and give simulation of the models. Section III gives details on the Cohen-Grossberg model of coupled neural networks and definition of synchronization both in the deterministic and stochastic case. We conclude with the summary.

II. Izhikevich model of neural network

The neuron is an electrically excitable cell that can be divided into three functionally distinct parts, called dendrites, soma, and axon. The dendrites receive signals from other neurons and pass them on to the soma. The soma performs a processing step of the inputs and if the total input exceeds a certain threshold, an output signal (called the action potential or spike) is generated. The action potential is then propagated down the axon and delivered to other neurons, causing a change in their membrane potentials. Communication among neurons occurs at gap junctions called synapses.

We will model the neurons as a differential equation of the type

$$\begin{aligned}\dot{\mathbf{x}}(t) &= \mathbf{b}(\mathbf{x}(t))\\ \mathbf{x}(0) &= x_0\end{aligned}$$

where $x_0 \in \mathbb{R}^d$, $\mathbf{b} : \mathbb{R}^d \to \mathbb{R}^d$ is a smooth vector field and the solution is the trajectory $\mathbf{x}(\cdot) : [0, \infty) \to \mathbb{R}^d$. The Izhikevich neuron is a two-dimensional system of ordinary differential equations of the form

$$\begin{aligned}\frac{\mathrm{d}V(t)}{\mathrm{d}t} &= 0.04V(t)^2 + 5V(t) - u(t) + I(t)\\ \frac{\mathrm{d}u(t)}{\mathrm{d}t} &= a(bV(t) - u(t)) \qquad (1)\end{aligned}$$

with the auxiliary after-spike resetting

$$V \leftarrow c \tag{2}$$

$$u \leftarrow u + d \tag{3}$$

The variable V represents the membrane potential and u represents the membrane recovery variable. Variables a, b, c, d describe the time scale of the recovery variable, sensitivity of u to subthreshold fluctuations of the membrane potential, the after-spike reset value of the membrane potential and the after-spike reset of the recovery variable, respectively.

We show a trajectory of the system (1) in Figure 1 by solving the differential equations via Euler's method. The trajectories of ordinary differential equations, however, look "smooth", which is contrast with empirical measurements of the membrane voltage in neurons. To model the observed noise, one adds a stochastic process (in our case we add the Brownian motion) to the ordinary differential equation, making it a stochastic differential equation (SDE)

$$\begin{aligned}\mathrm{d}\mathbf{X}(t) &= \mathbf{b}(\mathbf{X}(t))\mathrm{d}t + \mathbf{B}(\mathbf{X}(t))\mathrm{d}\mathbf{W}(t)\\ \mathbf{X}(0) &= x_0\end{aligned}$$

where $\mathbf{B} : \mathbb{R}^d \to \mathbb{M}^{d\times m}$ (the space of all $d \times m$ matrices) and $\mathbf{W}(t)$ is an m-dimensional Brownian motion. The stochastic Izhikevich neuron has the following form

$$\begin{aligned}\mathrm{d}V(t) &= (0.04V^2(t) + 5V(t) - u(t) + I(t))\mathrm{d}t + \mathrm{d}W(t)\\ \frac{\mathrm{d}u(t)}{\mathrm{d}t} &= a(bV(t) - u(t)) \qquad (4)\end{aligned}$$

We show a trajectory of the system in Figure 2 using the Euler-Maruyama method for numerical solution of SDEs.

A. Coupling of noisy Izhikevich neurons in a random network

We denote a network of N spiking inhibitory Izhikevich neurons by a weighted adjacency matrix $\mathbf{A} = [a_{ij}]_{i,j=1}^{N}$ where a_{ij} is the connection strength from neuron j to neuron i. We

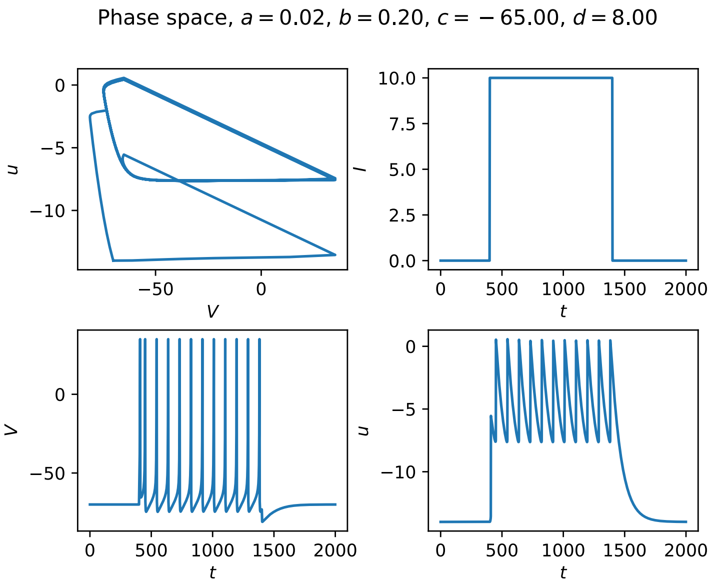


Fig. 1. A numerical solution to the two-dimensional system (1) via Euler's method with $\Delta t = 0.5$ with the parameters set to $a = 0.02, b = 0.2, c = -65, d = 8$. The top left subplot represents the phase space. Top right plots the injected current $I(t)$. The bottom left and right give the trajectory of the membrane potential and the recovery variable. As we can see, for this choice of parameters the system produces oscillations throughout the stimulation time.

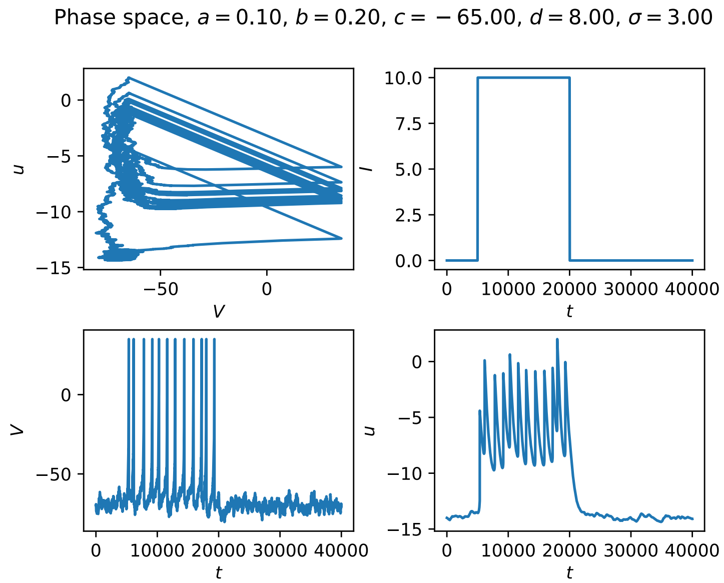


Fig. 2. A numerical solution to the stochastic two-dimensional system (4) via the Euler-Maruyama method for SDEs with $\Delta t = 0.01$ with the parameters set to $a = 0.1, b = 0.2, c = -65, d = 8$ and $\sigma = 3$.

model the synapses $\mathbf{R}(t) = [r_{ij}(t)]_{i,j=1}^N$ by exponential decay r_{ij},

$$\dot{r}_{ij} = \frac{-r_{ij}}{\tau_s} \tag{5}$$

and additionally, if the pre-synaptic neuron spikes, r_{ij} is updated as follows

$$r_{ij} \longleftarrow r_{ij} + 1 \tag{6}$$

The dynamics of the neurons are given by the following SDEs:

$$\begin{aligned} \mathrm{d}\mathbf{V}(t) &= \big(0.04\mathbf{V}^2(t) + 5\mathbf{V}(t) - \mathbf{u}(t) + \mathbf{I}^{\mathrm{syn}}(t) + I_0\big)\mathrm{d}t + \mathrm{d}\mathbf{W}(t) \\ \frac{\mathrm{d}\mathbf{u}(t)}{\mathrm{d}t} &= a(b\mathbf{V}(t) - \mathbf{u}(t)) \end{aligned} \tag{7}$$

where $\mathbf{V}(t) = (V_1(t), \dots, V_N(t))$ is the vector of membrane voltages for the N neurons, $\mathbf{u} = (u_1(t), \dots, u_N(t))$ are the recovery variables and $\mathbf{I}^{\mathrm{syn}}(t) = (I_1^{\mathrm{syn}}(t), \dots, I_N^{\mathrm{syn}}(t))$ is the vector of coupling mechanisms, which is given by the following equation

$$\mathbf{I}^{\mathrm{syn}}(t) = -\mathbf{A}\mathbf{R}(t)(\mathbf{V}(t) - E_{\mathrm{inh}}) \tag{8}$$

Here $E_{\mathrm{inh}} = -80$ mV is the reversal potential for an inhibitory synapse. We do not include electrical coupling, so as to inspect the effects of it's removal.

B. Synchronization

Next we define the synchronization measure S as introduced in [19] and use it to inspect the synchronization of spike trains throughout the stimulation time by a DC current.

First we compute the time fluctuations of the average membrane potential by

$$\Delta_N = \langle A_N(t)^2 \rangle_t - \langle A_N(t) \rangle_t^2 \tag{9}$$

where $\langle \cdot \rangle_t$ denotes the average over time and

$$A_N(t) = \frac{\sum_{i=1}^N V_i(t)}{N} \tag{10}$$

is the average membrane potential at time t. Subsequently, the population-averaged variance of the activity of each individual neuron is determined according to

$$\Delta = \frac{1}{N} \sum_{i=1}^N \left(\langle V_i(t)^2 \rangle_t - \langle V_i(t) \rangle_t^2 \right) \tag{11}$$

The synchronization measure is computed as

$$S = \frac{\Delta_N}{\Delta} \tag{12}$$

We have written a simulator in the Python language for the simulation of recurrent spiking neuronal networks coupled via dynamic synapses. Both deterministic and stochastic modes are available; we use the standard Euler method for the former and the Euler-Maruyama scheme for the latter mode. We ran simulations on networks with 100 inhibitory Izhikevich neurons for different parameters, see Figure 3. We noticed that when $\sigma = 0$, and controlling for a, b, c, d, increased spike transmission delay significantly affects spike synchronization. However, for sufficiently large noise $\sigma = 0.1$, the spikes did not synchronize. The removal of electrical coupling significantly disrupts the synchronization, whereas the model proposed in [14] that includes electrical coupling sustains its synchronization even after noise is added. This shows that it is important to understand the model and study its parameters, in order to give sufficient conditions for synchronization of the system. As we will see in the next section, such analysis exists for mathematical models of artificial neural networks.

III. Cohen-Grossberg model of coupled neural network

An artificial neural network consists of functionally connected processing units called neurons, which are modeled by some characteristics of biological neurons. The processing units are connected with signal channels called interconnections, which are all assigned a weight parameter. These parameters change through the learning process of the network. Similar as in biological neural networks, the learning in artificial neural networks happens by adjustment of their weight matrix, which stores the information. The neurons in the network can be organized in more layers (apart from the input layer for data feed and the output layer). The output information of each neuron depends on the input data supplied by the signal channels from the rest of the network, a threshold value of the neuron, and an activation function characteristic for the whole network.

There are different models of artificial neural networks, depending on the network architecture, the neural model and the learning strategy. We consider the model of coupled neural network, based on the original paper of Cohen and Grossberg [10] published in 1983. The model is given with the following system

$$\begin{gathered} dX_i(t) = a_i(X_i)\Big(b_i(X_i) - \sum_{j=1}^n c_{ij} d_j(X_j)\Big)\, dt, \\ \text{for} \quad t \geq 0,\, i = \overline{1,n} \quad \text{and some innitial condition,} \end{gathered} \tag{13}$$

where $X = (X_1, X_2, ..., X_n)$ is the state vector, $a_i(X_i)$ is an amplification function, $b_i(X_i)$ and $d_i(X_i)$ are self-signal and other-signal functions, and c_{ij} are interaction coefficients. This model is the generalization of many famous models, and has gained popularity due to its application on problems of associative memory, parallel computation and nonlinear optimization.

There are many reasons to introduce stochastic elements in the model - input data very often come from a system

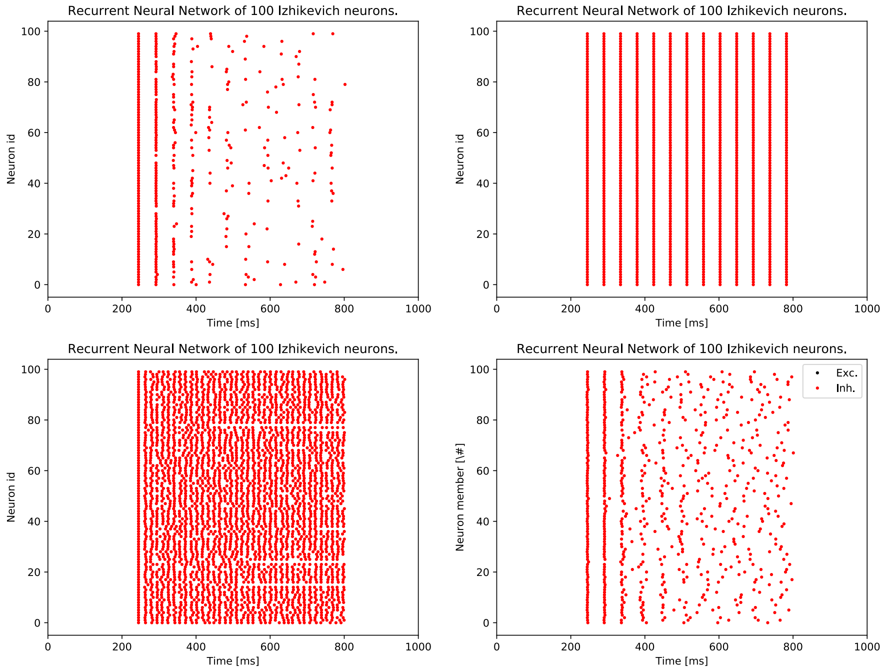


Fig. 3. Simulation of a neuronal network of 100 Izhikevich neurons. Top left: raster plot with $a = 0.02, b = 0.2, c = -65, d = 2$, injected current $I_0 = 4.1$, connection strength $w_{ij} = w = 0.1$, $\sigma = 0$, and $\tau_s = 0.7$. We see a synchronized firing pattern in the first two rounds, however, they break down after, with the synchronization measure being $S = 0.55$. Top right: same parameter configurations except that $\tau_s = 0.05$. We see a synchronized firing pattern throughout the stimulation time, with the synchronization measure here being 0.9. Bottom left: $a = 0.02, b = 0.2, c = -65, d = 2$, $\tau_s = 0.02$, $S = 0.3$. $\sigma = 0.01$. Bottom right: $a = 0.02, b = 0.2, c = -65, d = 2, w = 0.1, \tau_s = 0.1, \sigma = 0.01$, $S = 0.2$.

with noise and thus the input-output data, as well as the other variables of the networks, can be seen as a realization of a stochastic process. Also many scientists have emphasized the stochastic properties of the human brain [27, 31], seen even as a stochastic dynamical system [24]. We will consider the stochastic version of this model, given with the following coupled SDE

$$\begin{aligned} dX_i(t) \quad &= \quad a_i(X_i(t))\Big(b_i(X_i(t)) - \sum_{j=1}^{n} c_{ij} d_j(X_j(t))\Big)\, dt, \\ &\quad + \sigma(t, X_i(t))dW_i(t) \quad t \geq 0,\ i = \overline{1, n}, \end{aligned} \tag{14}$$

and some initial condition. Here $\sigma_i(t, X_i(t))$ are diffusion coefficients and $W(t) = (W_1, W_2, ..., W_n)$ is n-dimensional Brownian motion. Next we give the definition for synchronization of systems (13) and (14) respectively

Definition 1:

- [7] The coupled system (13) is said to be globally exponentially synchronized if for each $\varepsilon > 0$ there exists a constant $M > 0$, s.t. for all initial conditions, for sufficiently large $T > 0$ we have

$$||X_i(t) - X_j(t)|| \leq M \exp^{-\varepsilon t} \tag{15}$$

  where $|| \cdot ||$ is the Euclidean norm.

- [9] The coupled stochastic system (14) is said to be p-moment exponentially synchronized if for any initial condition, there exist two positive constants M, ε s.t.

$$\mathbb{E}||X_i(t) - X_j(t)||^p \leq M \exp^{-\varepsilon t}, t > 0 \tag{16}$$

  where $|| \cdot ||$ is the p-norm.

The analysis of synchronization of these systems is complicated, especially when we add additional conditions like time delay, impulsive effects, Markowian switching, complex coupling architecture etc. This asks for a detailed study to understand the influence of the model parameters on the synchronization and is left for our future work. We refer to the broad literature on this topic, for example [7, 8, 25] and the references therein.

IV. Summary

In this paper we have presented the problem of synchronization of coupled neural networks. We have defined and illustrated the problem on the Izhikevich model of neurons and we introduced the problem in the case of coupled Cohen-Grossberg neural network. We have included a lot of references on different problems which may be analyzed in our future work.

References

[1] L. F. Abbott and C. van Vreeswijk. Asynchronous states in networks of pulse-coupled oscillators. *Phys. Rev. E*, 48:1483–1490, Aug 1993.

[2] Nikolai Axmacher, Florian Mormann, Guillen Fernández, Christian E Elger, and Juergen Fell. Memory formation by neuronal synchronization. *Brain research reviews*, 52(1):170–182, 2006.

[3] Wyeth Bair and Christof Koch. Temporal precision of spike trains in extrastriate cortex of the behaving macaque monkey. *Neural computation*, 8(6):1185–1202, 1996.

[4] Christoph Börgers and Nancy Kopell. Effects of noisy drive on rhythms in networks of excitatory and inhibitory neurons. *Neural computation*, 17(3):557–608, 2005.

[5] P. C. Bressloff. Mean-field theory of globally coupled integrate-and-fire neural oscillators with dynamic synapses. *Physical Review E*, 60(2):2160, 1999.

[6] György Buzsáki and Andreas Draguhn. Neuronal oscillations in cortical networks. *science*, 304(5679):1926–1929, 2004.

[7] J. Cao, P. Li, and W. Wang. Global synchronization in arrays of delayed neural networks with constant and delayed coupling. *Physics Letters A*, 353(4):318–325, 2006.

[8] G. Chen, J. Zhou, and Z. Liu. Global synchronization of coupled delayed neural networks and applications to chaotic CNN models. *International Journal of Bifurcation and Chaos*, 14(07):2229–2240, 2004.

[9] Zhang Chen. Complete synchronization for impulsive cohen–grossberg neural networks with delay under noise perturbation. *Chaos, Solitons Fractals*, 42(3):1664 – 1669, 2009.

[10] M. A. Cohen and S. Grossberg. Absolute stability of global pattern formation and parallel memory storage by competitive neural networks. *IEEE Trans. Syst., Man, and Cybern.*, 13(5):815–826, 1983.

[11] Gustavo Deco, Edmund T Rolls, and Ranulfo Romo. Stochastic dynamics as a principle of brain function. *Progress in neurobiology*, 88(1):1–16, 2009.

[12] Ilan Dinstein, Karen Pierce, Lisa Eyler, Stephanie Solso, Rafael Malach, Marlene Behrmann, and Eric Courchesne. Disrupted neural synchronization in toddlers with autism. *Neuron*, 70(6):1218–1225, 2011.

[13] Wulfram Gerstner and Werner M Kistler. *Spiking neuron models: Single neurons, populations, plasticity*. Cambridge university press, 2002.

[14] Daqing Guo, Qingyun Wang, and Matjaž Perc. Complex synchronous behavior in interneuronal networks with delayed inhibitory and fast electrical synapses. *Physical Review E*, 85(6):061905, 2012.

[15] Eugene M Izhikevich. Simple model of spiking neurons. *IEEE Transactions on neural networks*, 14(6):1569–1572, 2003.

[16] Zachary F Mainen and Terrence J Sejnowski. Reliability of spike timing in neocortical neurons. *Science*, 268(5216):1503–1506, 1995.

[17] Lucia Melloni, Carlos Molina, Marcela Pena, David Torres, Wolf Singer, and Eugenio Rodriguez. Synchronization of neural activity across cortical areas correlates with conscious perception. *Journal of neuroscience*, 27(11):2858–2865, 2007.

[18] R. E. Mirollo and S. H. Strogatz. Synchronization of pulse-coupled biological oscillators. *SIAM Journal on Applied Mathematics*, 50(6):1645–1662, 1990.

[19] L Neltner, David Hansel, Germán Mato, and Claude Meunier. Synchrony in heterogeneous networks of spiking neurons. *Neural computation*, 12(7):1607–1641, 2000.

[20] Ashok Patel and Bart Kosko. Stochastic resonance in noisy spiking retinal and sensory neuron models. *Neural Networks*, 18(5-6):467–478, 2005.

[21] L. M. Pecora and T. L. Carroll. Master stability functions for synchronized coupled systems. *Physical Review Letters*, 80(10):2109–2112.

[22] T. Pereira. Stability of synchronized motion in complex networks (lecture notes). *arXiv preprint arXiv:1112.2297*, 2011.

[23] Benjamin Pfeuty, Germán Mato, David Golomb, and David Hansel. The combined effects of inhibitory and electrical synapses in synchrony. *Neural Computation*, 17(3):633–670, 2005.

[24] T. J. Sejnowski. Skeleton filters in the brain. In Geoffrey E. Hinton and James A. Anderson, editors, *Parallel Models of Associative Memory*, pages 189–212. Erlbaum, Hillsdale, NJ, USA, 1981.

[25] Q. Song. Synchronization analysis of coupled connected neural networks with mixed time delays. *Neurocomputing*, 72(16):3907–3914, 2009.

[26] David Sterratt, Bruce Graham, Andrew Gillies, and David Willshaw. *Principles of computational modelling in neuroscience*. Cambridge University Press, 2011.

[27] J. G. Taylor. Spontaneous behaviour in neural networks. *Journal of Theoretical Biology*, 36(3):513 – 528, 1972.

[28] Gašper Tkačik, Jason S Prentice, Vijay Balasubramanian, and Elad Schneidman. Optimal population coding by noisy spiking neurons. *Proceedings of the National Academy of Sciences*, 107(32):14419–14424, 2010.

[29] Nicolaas Godfried Van Kampen. *Stochastic processes in physics and chemistry*, volume 1. Elsevier, 1992.

[30] Rob R de Ruyter van Steveninck, Geoffrey D Lewen, Steven P Strong, Roland Koberle, and William Bialek. Reproducibility and variability in neural spike trains. *Science*, 275(5307):1805–1808, 1997.

[31] J. von Neumann. *The Computer and the Brain*. Yale University Press, New Haven, CT, USA, 1958.

[32] John A White, Matthew I Banks, Robert A Pearce, and Nancy J Kopell. Networks of interneurons with fast and slow γ-aminobutyric acid type a (gabaa) kinetics provide substrate for mixed gamma-theta rhythm. *Proceedings of the National Academy of Sciences*, 97(14):8128–8133, 2000.

Transforming Geospatial RDF Data into GeoSPARQL-Compliant Data: A Case of Traffic Data

Milos Jovanovik
Faculty of Computer Science and Engineering
Ss. Cyril and Methodius University
Skopje, North Macedonia
milos.jovanovik@finki.ukim.mk

Mirko Spasić
Faculty of Mathematics
University of Belgrade
Belgrade, Serbia
mirko@matf.bg.ac.rs

Abstract—**Geospatial RDF datasets have a tendency to use latitude and longitude properties to denote the geographic location of the entities described within them. On the other hand, geographic information systems prefer the use of WKT and GML geometries when working with geospatial data. In this paper, we present a process of RDF data transformation which produces a GeoSPARQL-compliant dataset, using an RDF geospatial dataset with traffic data as a starting point. The traffic is comprised of vehicle traces, which consist of numerous points with specific latitude and longitude values. With our transformations, we enable querying of the dataset with GeoSPARQL extensions, which can be used to feed a GIS solution.**

Index Terms—**GeoSPARQL, Geospatial Data, Data Transformation, GIS, RDF, SPARQL, Linked Data**

I. Introduction

The exploration gain of semantic technology for spatial data management becomes more and more evident in many different domains. Specific taxonomies and ontologies have been used on thematic web portals for representing a variety of categories with geospatial features. With the increasing popularity of this technology within the spatial domain, the need for standardization is growing. Initial efforts include the Basic Geo Vocabulary[1] by the World Wide Web Consortium (W3C), which provides a namespace for representing latitude, longitude and other information about geospatial entities using WGS84 as a standard of the coordinate system. Later achievements comprise GeoRSS[2], GeoOWL[3], NeoGeo Geometry Ontology[4], GeoJSON[5], GeoRDF[6], etc. Finally, GeoSPARQL has emerged as a promising standard from W3C for geospatial RDF. It supports both representation and querying of geospatial data on the Semantic Web and defines a new vocabulary. It suggests a concrete ontology for representing features and geometries in RDF as Well Known Text (WKT) or Geography Markup Language (GML) literals. GeoSPARQL defines a core set of classes, properties and data types that can be used to construct query patterns in an extension of SPARQL. Its main aim is to ensure a consistent representation of geospatial semantic data across the Web, thus allowing both vendors and users to achieve uniform access to geospatial RDF data.

Although the standard is sufficiently mature, now being 7 years old, the data providers are not familiar enough with it, or more often they take the path of least resistance and publish their data in their own format or use custom ontologies. In these scenarios, an additional effort is needed in order to understand their data, and use it within an application. This way, cross-domain data mixing and fusion is very complicated and almost impossible. A better approach is to develop a means for transforming the dataset in question into a GeoSPARQL-compliant one, ready for use by a large number of stakeholders who already have knowledge of the standard. That was the main goal of our transformations described in this paper.

II. Related Work

Transforming data from one format to the other – whether related to a simple conversion between different file formats, or more complicated mappings from relational data to linked data, or even generation of datasets following different standards – is a fundamental aspect of most data integration and data management tasks. It can vary from very simple or very complex, depending on the required changes to the data and their model. This is an area that attracts the attention of many data scientists for very practical reasons. Integrating data from heterogeneous sources has led to the development of Extract-Transform-Load (ETL) systems and methodologies, as a means of addressing modern interoperability challenges.

One group of these are general tools for converting relational data from traditional DBMSs to the RDF data model, such as Triplify [1], D2R Server[7], or Virtuoso RDFizer Middleware (Sponger)[8]. On the other hand, there are libraries

[1] http://www.w3.org/2003/01/geo/

[2] http://www.georss.org/

[3] https://www.w3.org/2005/Incubator/geo/XGR-geo-20071023/W3C_XGR_Geo_files/geo_2007.owl

[4] http://geovocab.org/

[5] http://geojson.org/

[6] https://www.w3.org/wiki/GeoRDF

[7] http://d2rq.org/d2r-server

[8] http://docs.openlinksw.com/virtuoso/virtuososponger.html

and tools which deal with geospatial data, but without any RDF support: GDAL/OGR[9], which is a translator library for raster and vector geospatial data formats, GeoKettle[10], which is a spatial ETL tool dedicated to the integration of different spatial data sources for building and updating geospatial data warehouses, etc. It is worth to mention geometry2rdf[11], as a library for generating RDF from resources with geometrical information, but its RDF model is not compliant with the GeoSPARQL standard.

The largest open geospatial dataset is Linked Geo Data, a spatial knowledge base which has been derived from Open Street Map[12]. The authors of [2] elaborate on how the collaboratively collected OSM data can be interactively transformed and represented adhering to the RDF data model, building a rich integrated and interlinked geographic dataset for the Semantic Web. Their approach is to store the mapping information in a relational database (PostGIS) together with the OSM data, and use the SPARQL-to-SQL rewriter Sparqlify[13] to generate RDF. The mapping uses the GeoSPARQL vocabulary to represent the data.

TripleGeo [3] is an open-source tool that can produce geospatial datasets in compliance with GeoSPAQL standard, based on different inputs. It is an ETL utility that can extract geospatial features from various sources and transform them into triples for subsequent loading into RDF stores. TripleGeo can directly access both geometric representations and thematic attributes either from standard geographic formats or widely used DBMSs. It can also reproject input geometries on-the-fly into a different coordinate reference system, before exporting the resulting triples into a variety of notations.

III. Traffic Dataset

As part of our activities in the project "SAGE: Semantic Geospatial Analytics"[14] [4], we use traffic datasets generated by the synthetic data generator from TomTom [5]. The synthetic data generator can generate RDF datasets containing synthetic traces of vehicles on public roads, based on a specialized algorithm which mimics real-world traffic.

The datasets contain `Trace` entities, representing a one-day trace of a vehicle on a map, where each `Trace` consists of multiple `Point` entities. Each `Point` has a `latitude`, a `longitude`, a `timestamp` and a `Speed` entity, depicting a specific position in time and space of the vehicle. The `Speed` entity has a velocity `value` and `metric`.

The size of the generated dataset can be specified in advance, by specifying the number of traces we want it to contain. For the research presented in this manuscript, we generated a dataset which contains 1,000,000 traces of vehicles in and around the city of Leipzig, spanning from the beginning of year 2016 to the end of 2017. These `Trace` contain a total of over 126,000,000 `Point` entities. In total, the synthetic RDF dataset contains over 889,000,000 RDF triples.

IV. GeoSPARQL Enhancements of the Dataset

The synthetic dataset uses a very basic ontology, depicting the classes and properties mentioned above. The produced dataset is in RDF, but it is not GeoSPARQL compliant, i.e. it does not contain `Feature` and `Geometry` instances, which have explicit well-known-text (WKT) geometries [6]. This means that while the dataset itself is a geospatial dataset, it cannot be directly used with GeoSPARQL and in applications which require explicit WKT values for the geospatial entities.

To improve this, and enable application interactions with the dataset with GeoSPARQL queries, we created general enhancements for the traffic datasets. With them, we produce GeoSPARQL-enabled datasets, suitable for the project, its use-cases and general compliance testing.

The GeoSPARQL enhancements are defined as SPARQL queries which can be executed over a TomTom synthetic traffic dataset which is loaded into an RDF triplestore [7]. The enhancements include `Trace` transformations, `Point` transformations, and some additional transformation necessary for improved use-cases.

A. Trace Transformations

The trace transformations have the purpose of creating a LineString geometry for each trace in the dataset. Since the dataset model is specific for the data generator we use to generate the datasets, we need a custom transformation which will select the latitude and longitude values of all points which comprise a given trace, and construct the corresponding WKT LineString geometry [7]. To achieve this, we use the following transformations for each trace:

- Each `Trace` is enhanced to represent a `geo:Feature` entity, which has a `geo:hasGeometry` relation with a geometry entity specified as a `sf:LineString` entity;
- The `sf:LineString` entity has a `geo:asWKT` relation to a `geo:wktLiteral` value, which represents the entire `Trace` as a GeoSPARQL `LINESTRING`.

The SPARQL INSERT query for the trace transformations adds new triples into the dataset graph, via the snippet below.

Trace Enhancements Snippet

```
INSERT {
    ?trace rdf:type geo:Feature .
    ?trace geo:hasGeometry ?traceGeomID .
    ?traceGeomID a sf:LineString .
    ?traceGeomID geo:asWKT ?wkt .
}
```

Here, `geo:Feature` and `sf:LineString` are classes defined by the GeoSPARQL standard [6]. On the other hand, `?traceGeomID` is constructed by a simple string concatenation between the original URI of the trace and a fixed suffix. The `?wkt` value is constructed by reading all points from the trace in a subquery, ordering them by their timestamp,

[9] http://www.gdal.org/

[10] http://www.spatialytics.org/projects/geokettle/

[11] https://github.com/boricles/geometry2rdf

[12] https://www.openstreetmap.org/about

[13] http://aksw.org/Projects/Sparqlify.html

[14] http://sageproject.eu/

constructing a combined string of the latitude and longitude of each point, and then combining all concatenated latitudes and longitudes of the ordered points in a format which defines the complete trace as a LineString. An example WKT LineString value for a trace is shown below.

WKT LineString Example

```
LINESTRING(
    12.127818 51.284445,
    12.124343 51.282212,
    12.120798 51.280137,
    12.117422 51.277686,
    12.114321 51.275172,
    12.111059 51.272767,
    ...
    12.174391 51.328762)
```

B. Point Transformations

Similarly as with traces, we aimed to transform each point of each trace into a Point geometry [7]. Therefore, for each point we use the following transformations:

- Each `Point` is enhanced to represent a `geo:Feature` entity, which has a `geo:hasGeometry` relation with a geometry entity specified as a `sf:Point` entity;
- The `sf:Point` entity has a `geo:asWKT` relation to a `geo:wktLiteral` value, which represents the `Point` as a GeoSPARQL `POINT`.

Similarly as with the traces, the SPARQL INSERT query for the point transformations also adds new triples into the dataset graph, via the snippet below.

Point Enhancements Snippet

```
INSERT {
    ?point rdf:type geo:Feature .
    ?point geo:hasGeometry ?pointGeomID .
    ?pointGeomID a sf:Point .
    ?pointGeomID geo:asWKT ?wkt .
}
```

As with the traces, the `geo:Feature` and `sf:Point` classes used in the INSERT clause are classes defined by the GeoSPARQL standard, while `?pointGeomID` is constructed by a simple string concatenation between the original URI of the point and a fixed suffix. The `?wkt` value is constructed by reading and concatenating the latitude and longitude values of the point, in a format which defines a correct Point. An example WKT Point value is shown below.

WKT Point Example

```
POINT(12.127818 51.284445)
```

C. Additional Transformations

Aside from the trace and point transformations, we made additional transformations which are not directly related to GeoSPARQL, but enhance the use-cases [7]:

- Adding a numerical ID to each `Trace` entity, via a new property: `traces:numID`;
- Adding an explicit relation between a `Trace` and its start and end points, via new properties: `traces:hasStartPoint` and `traces:hasEndPoint`, respectively;
- Adding an explicit relation between a `Trace` and its calculated duration, in seconds, via a new property: `traces:hasDuration`.

After running all enhancements over our Leipzig traffic dataset, we ended up with over 1,532,000,000 RDF triples in total in the dataset. This dataset is available within a Virtuoso instance [8], made available as part of the SAGE project.

V. Using the GeoSPARQL-Compliant Dataset

In order to demonstrate the usability of the resulting GeoSPARQL-compliant dataset, we present several use-cases in the form of SPARQL queries, query results and their actual usage within a geographic information system. For brevity, all prefix definitions and FROM clauses of the SPARQL queries have been omitted from the examples, but can be seen in full detail on the GitHub page of the project [7]. All examples below use our enhanced synthetic traffic dataset for the city of Leipzig, described previously.

1) Query 1: Find all vehicle traces started within a specified time period and select their respective WKT `LINESTRING` values, to be drawn on the map. Additionally, select the duration of each such trace, and calculate the distance between the starting point and the ending point of the trace.

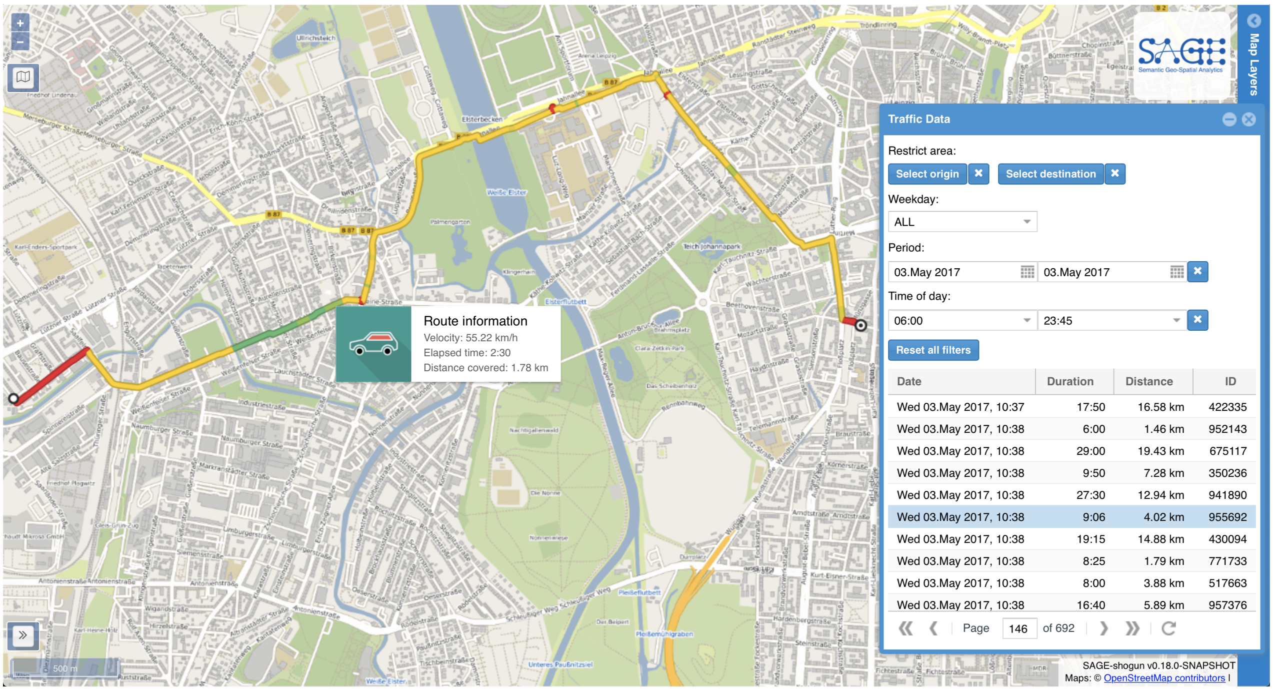
SPARQL Query 1

```
SELECT ?wkt ?date ?duration ?distance ?traceID
WHERE {
    ?trace a traces:Trace ;
        geo:hasGeometry ?traceGeom ;
        traces:numID ?traceID ;
        traces:hasDuration ?duration ;
        traces:hasStartPoint ?start ;
        traces:hasEndPoint ?end .
    ?traceGeom geo:asWKT ?wkt .
    ?start traces:hasTimestamp ?date .

    FILTER (?date >=
        "2017-05-03T06:00:00Z"^^xsd:dateTime
            && ?date <=
        "2017-05-03T23:45:00Z"^^xsd:dateTime)

    ?start geo:hasGeometry ?startGeom .
    ?startGeom geo:asWKT ?startWKT .
    ?end geo:hasGeometry ?endGeom .
    ?endGeom geo:asWKT ?endWKT .

    BIND(geof:distance(?startWKT,
                ?endWKT,
                units:meter) as ?distance)
} ORDER BY ?date
```

This query returns results for all traces of vehicles which started their route in the specified time period. A partial result

TABLE I
PARTIAL RESULTS FROM QUERY 1.

Trace as WKT	Date and Time	Duration (s)	Distance (m)	Trace ID
LINESTRING(12.353488 51.343931,12.352586 51.343744, ...	2017-05-03T08:37:59Z	1,070	16,581.50	422335
LINESTRING(12.610297 51.441528,12.612627 51.441695, ...	2017-05-03T08:38:11Z	360	1,459.58	952143
LINESTRING(12.491782 51.381629,12.491827 51.381999, ...	2017-05-03T08:38:19Z	1,740	19,425.20	675117
LINESTRING(12.207745 51.407249,12.207753 51.406779, ...	2017-05-03T08:38:21Z	590	7,281.98	350236
LINESTRING(12.489270 51.272567,12.491189 51.275444, ...	2017-05-03T08:38:41Z	1,650	12,941.70	941890
LINESTRING(12.315006 51.328223,12.315075 51.328165, ...	2017-05-03T08:38:43Z	546	4,021.36	955692
LINESTRING(12.409652 51.335220,12.408825 51.335428, ...	2017-05-03T08:38:52Z	1,155	14,875.00	430094
LINESTRING(12.528184 51.267206,12.528220 51.267318, ...	2017-05-03T08:38:57Z	505	1,786.52	771733
LINESTRING(12.362608 51.358667,12.362919 51.359149, ...	2017-05-03T08:38:58Z	480	3,876.72	517663
LINESTRING(12.127818 51.284445,12.124343 51.282212, ...	2017-05-03T08:38:58Z	1,000	5,892.86	957376



Route information
Velocity: 55.22 km/h
Elapsed time: 2:30
Distance covered: 1.78 km

Traffic Data

Restrict area:

Weekday: ALL

Period: 03.May 2017 03.May 2017

Time of day: 06:00 23:45

Date	Duration	Distance	ID
Wed 03.May 2017, 10:37	17:50	16.58 km	422335
Wed 03.May 2017, 10:38	6:00	1.46 km	952143
Wed 03.May 2017, 10:38	29:00	19.43 km	675117
Wed 03.May 2017, 10:38	9:50	7.28 km	350236
Wed 03.May 2017, 10:38	27:30	12.94 km	941890
Wed 03.May 2017, 10:38	9:06	4.02 km	955692
Wed 03.May 2017, 10:38	19:15	14.88 km	430094
Wed 03.May 2017, 10:38	8:25	1.79 km	771733
Wed 03.May 2017, 10:38	8:00	3.88 km	517663
Wed 03.May 2017, 10:38	16:40	5.89 km	957376

Page 146 of 692

SAGE-shogun v0.18.0-SNAPSHOT
Maps: © OpenStreetMap contributors |

Fig. 1. Visual representation of the results of Query 1: mapping them within a GIS solution.

is shown in Table I, and the same traces are depicted on Figure 1 as part of a GIS solution which visualizes one of the selected WKT traces. The difference in the date-time values between the table and the figure is due to different time-zones: the dataset holds the date-time values in GMT time, as they are originally generated, while the GIS application transforms and displays them into CET time, for user-experience purposes. Note that the GIS application sends out an additional SPARQL query to the dataset to get the speed values of each point of the selected trace, the result of which is visible in Figure 1 via the trace coloring. For brevity, this additional query is omitted from the example.

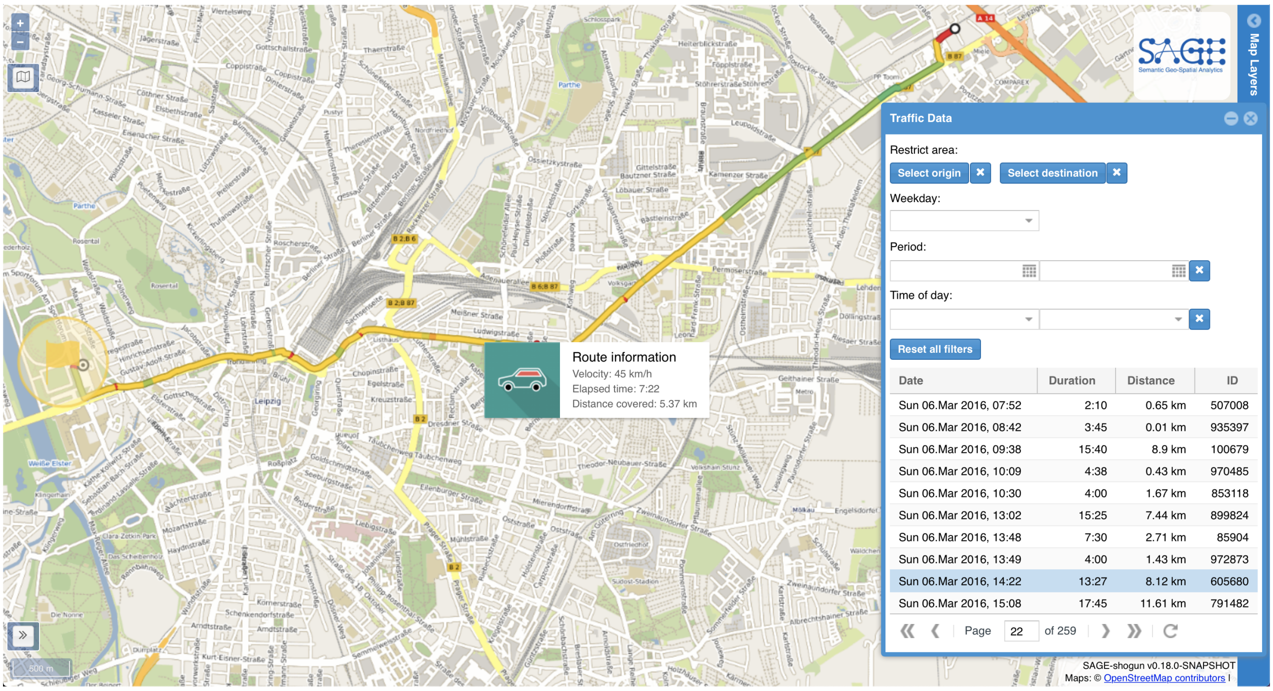
2) *Query 2:* Find all vehicle traces which have a given map region as a destination. The query selects all traces which satisfy the constraints, gets their WKT `LINESTRING` values, their duration and calculates the distance between the starting point and the ending point of the trace.

SPARQL Query 2

```
SELECT ?wkt ?date ?duration ?distance ?traceID
WHERE {
    ?trace a traces:Trace ;
        geo:hasGeometry ?traceGeom ;
        traces:numID ?traceID ;
        traces:hasDuration ?duration ;
        traces:hasStartPoint ?start ;
        traces:hasEndPoint ?end .
    ?traceGeom geo:asWKT ?wkt .
    ?start traces:hasTimestamp ?date ;
        geo:hasGeometry ?startGeom .
    ?startGeom geo:asWKT ?startWKT .
    ?end geo:hasGeometry ?endGeom .
    ?endGeom geo:asWKT ?endWKT .

    FILTER(geof:sfContains(
      bif:ST_GeometryFromText("POLYGON((
        12.346821967457 51.34679532759,
        12.348366919850 51.34703657322,
```

TABLE II
PARTIAL RESULTS FROM QUERY 2.

Trace as WKT	Date and Time	Duration (s)	Distance (m)	Trace ID
LINESTRING(12.354888 51.349741,12.353951 51.349536, ...	2016-03-06T06:52:34Z	130	651.96	507008
LINESTRING(12.351120 51.345119,12.350997 51.344888, ...	2016-03-06T07:42:50Z	225	8.93	935397
LINESTRING(12.292396 51.413958,12.293035 51.413740, ...	2016-03-06T08:38:42Z	940	8,899.60	100679
LINESTRING(12.354429 51.348359,12.354407 51.348249, ...	2016-03-06T09:09:35Z	278	429.51	970485
LINESTRING(12.358288 51.360596,12.358529 51.360280, ...	2016-03-06T09:30:12Z	240	1,670.04	853118
LINESTRING(12.269029 51.305859,12.268899 51.306153, ...	2016-03-06T12:02:25Z	925	7,440.02	899824
LINESTRING(12.347528 51.370391,12.348132 51.371072, ...	2016-03-06T12:48:52Z	450	2.714.82	85904
LINESTRING(12.338960 51.331886,12.338704 51.331418, ...	2016-03-06T12:49:42Z	240	1,429.53	972873
LINESTRING(12.462456 51.370047,12.462360 51.370006, ...	2016-03-06T13:22:17Z	807	8,120.75	605680
LINESTRING(12.210354 51.399915,12.210412 51.399675, ...	2016-03-06T14:08:47Z	1,065	11,607.40	791482



Route information
Velocity: 45 km/h
Elapsed time: 7:22
Distance covered: 5.37 km

Traffic Data

Restrict area:
Select origin ✖ Select destination ✖
Weekday:
Period:
Time of day:
Reset all filters

Date	Duration	Distance	ID
Sun 06.Mar 2016, 07:52	2:10	0.65 km	507008
Sun 06.Mar 2016, 08:42	3:45	0.01 km	935397
Sun 06.Mar 2016, 09:38	15:40	8.9 km	100679
Sun 06.Mar 2016, 10:09	4:38	0.43 km	970485
Sun 06.Mar 2016, 10:30	4:00	1.67 km	853118
Sun 06.Mar 2016, 13:02	15:25	7.44 km	899824
Sun 06.Mar 2016, 13:48	7:30	2.71 km	85904
Sun 06.Mar 2016, 13:49	4:00	1.43 km	972873
Sun 06.Mar 2016, 14:22	13:27	8.12 km	605680
Sun 06.Mar 2016, 15:08	17:45	11.61 km	791482

« ‹ Page 22 of 259 › »
SAGE
Semantic Geo-Spatial Analytics
Map Layers
SAGE-shogun v0.18.0-SNAPSHOT
Maps: © OpenStreetMap contributors

Fig. 2. Visual representation of the results of Query 2: mapping them within a GIS solution.

```
      12.350898925160 51.34706337821,
      12.352486792897 51.34655408069,
      12.353817168568 51.34574991518,
      12.354503814076 51.34449002752,
      ...
      12.346821967457 51.34679532759))"),
    ?endWKT))

  BIND(geof:distance(?startWKT,
                     ?endWKT,
                     units:meter) as ?distance)
} ORDER BY ?date
```

In this example we select all vehicle traces which end on the parking grounds around the main football stadium in Leipzig. Part of the results are presented in Table II and on Figure 2, along with one selected trace which is displayed in full detail. The map region selected as the destination filter for traces can be seen in the left part of the figure, as a yellow circle with a yellow flag in the middle.

It is important to note that both of these examples, and their SPARQL queries, would not be possible without the introduced transformations of the original dataset. Namely, the SPARQL functions `geof:distance` and `geof:sfContains`, just like all other GeoSPARQL functions, cannot work with arguments which are not valid WKT values, such as our LineStrings and Points.

Many other examples and use-cases are made available with the transformation of the latitude and longitude coordinates into comprehensive WKT literals, and with the additional enhancements we did as part of the work presented here. For instance, some of the other GeoSPARQL function can be used to find intersections between regions on a map, e.g. a street, and the entire trace, not just its start and end point. This would provide a way to select vehicle traces which traverse a specific location on a map, on a particular day, in a particular time.

The examples shown here have been integrated as features of the GIS solution presented in Figures 1 and 2, where it generates and issues GeoSPARQL-extended SPARQL queries to the dataset. The GeoSPARQL-compliant RDF results are then used by the application in a convenient manner.

VI. Conclusion

The transformation of a simple geospatial RDF dataset into a GeoSPARQL-compliant dataset brings the data quality to a significantly high level, where it can be used by GIS software which does not speak RDF. This is a very important step for both providers and users of geospatial data as it allows interoperability between the already existing toolset for dealing with geospatial data on one hand, and the RDF dataset providers on the other hand. We demonstrate this by transforming a synthetic RDF dataset in the traffic domain into a dataset which can be queried using the GeoSPARQL extensions of SPARQL, and can be visualized in a GIS application which draws WKT geometries on a map.

Acknowledgment

This work has been supported by Eurostars Project SAGE (GA no. E!10882)

References

[1] S. Auer, S. Dietzold, J. Lehmann, S. Hellmann, D. Aumueller. "Triplify: Light-weight Linked Data Publication from Relational Databases", 18th International Conference on World Wide Web, pp. 621–630, April 2009.

[2] C. Stadler, J. Lehmann, K. Hffner, S. Auer, "LinkedGeoData: A Core for a Web of Spatial Open Data", Semantic Web Journal, vol. 3, no. 4, pp. 333–354, 2012.

[3] K. Patroumpas, M. Alexakis, G. Giannopoulos, S. Athanasiou, "TripleGeo: An ETL Tool for Transforming Geospatial Data into RDF Triples". EDBT/ICDT Workshops, pp. 275–278, 2014.

[4] M. Jovanovik, H. Williams, M. Spasić, "D4.1: Prototype Verifying Virtuoso GeoSPARQL and DE-9IM Compliance", Project deliverable from Project SAGE, January 2019.

[5] K. Bösche, T. Sellam, H. Pirk, R. Beier, P. Mieth, S. Manegold, "Scalable Generation of Synthetic GPS Traces with Real-Life Data Characteristics.", Technology Conference on Performance Evaluation and Benchmarking, pp. 140–155, August 2012.

[6] Open Geospatial Consortium, "OGC GeoSPARQL - A Geographic Query Language for RDF Data". Open Geospatial Consortium, September 2012. https://www.opengeospatial.org/standards/geosparql, Document 11-052r4.

[7] M. Jovanovik, "Traffic Data to GeoSPARQL" GitHub Project Page: https://github.com/mjovanovik/TrafficData-to-GeoSPARQL, Retrieved on 12 April 2019.

[8] O. Erling, "Virtuoso, a Hybrid RDBMS/Graph Column Store", IEEE Data Eng. Bull., vol. 35, no. 1, pp. 3–8, 2012.

SHORT PAPERS

Exploratory Analysis of Student Activities and Success Based On Moodle Log Data

Neslihan Ademi
Computer Engineering Department
International Balkan University
Skopje, North Macedonia
neslihan@ibu.edu.mk

Suzana Loshkovska
Faculty of Computer Science and Engineering
Ss. Cyril and Methodius University
Skopje, North Macedonia
suzana.loshkovska@finki.ukim.edu.mk

***Abstract*— This paper presents the effects of online engagements of the students with their achieved grades in a blended course environment. Exploratory data analysis of the Moodle logs help to understand the trends in students' usage of the learning system and its effect on their success. Data pre-processing, transformation, integration and statistical data analysis methods are used in RStudio environment for the future data mining experiments. It is found that all online activities of the students on Moodle have positive correlations with the course grade.**

Index Terms—*Learning Systems, Moodle logs, Data Mining, Engagement, Student Success*

I. Introduction

Course management systems (CMSs) offer a great variety of opportunities to facilitate information sharing and communication among participants in a course. They let educators distribute information to students, produce content material, prepare assignments and tests, engage in discussions, manage distance classes and enable collaborative learning with forums, chats, file storage areas, etc. One of the most commonly used CMS is Moodle (modular object-oriented developmental learning environment), which is a free learning management system. This type of e-learning systems contains a huge amount of data which enables analyzing students' behavior and creating educational data.

Most analyses of log data collected through observational studies provide a descriptive overview of human behavior. Simply observing behavior at scale provides insights about how people interact with existing systems and services, often revealing surprises [1]. Observational Log Studies contain two common ways to partition log data; by time and by user. Partitioning by time is interesting because log data often contains significant temporal features, such as periodicities (including consistent daily, weekly, and yearly patterns) and spikes in behavior during important events. It is often possible to get an up-to-the- minute picture of how people are behaving with a system from log data by comparing past and current behavior. It is also interesting to partition log data by user characteristics [1].

Recently there are many studies in the literature about log analysis in e-learning environments [2]–[7].

The e-learning data mining process consists of the same four steps in the general data mining process. Collect data, Pre-process the data, Apply data mining, Interpret, evaluate and deploy the results. [8]

The paper presents a case study on the analysis of Moodle log data by using data mining methods. The second section defines the used methodology for the analysis, while the third section gives results and discussion, finally the last section is the conclusion.

II. Methodology

In this section, step by step used methodology is explained. Fig. 1 gives the work flow these steps.

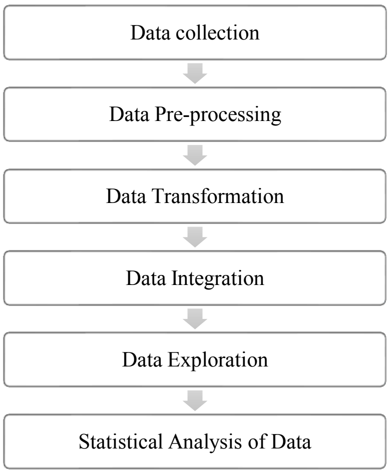


Fig. 1. Flow chart of the used methodology

A. Data Collection

For the study, log files are taken from Moodle which is installed and used at the Faculty of Computer Science and Engineering in the form of .csv containing all activities of the students from a one semester course of User Interfaces at the academic year 2016-2017.

The standard retrieved fields in the log files are: Time, User full name, Affected user, Event context, Component, Event name, Description, Origin, IP address

In our study the retrieved data was composed of 161007 rows, each with a filled value in every of the above-mentioned column fields. These data correspond to a one semester period for a bachelor's degree course User Interfaces at the Faculty of Computer Science and Engineering at the University of Ss. Cyril and Methodius in Skopje. Teaching process was designed as blended learning. Moodle was used to support classroom teaching to distribute course material, lectures, homework, laboratory exercises and to provide discussion through the forums. A total of 265 students registered with Moodle for the course of User Interfaces.

Separately, another file is used which contained scores and grades of the students from the course.

B. Data Pre-processing

To keep on relevant and correct information, pre-processing is applied to the log files. As we want to analyze only the students' actions; the actions logged by instructors and administrators were selectively removed. Log data produced by the system is also removed by filtering the data where component field is system. Required fields are extracted and duplicate records are removed.

Description field contained text. By extracting userID and moduleID from the description text we generated new columns for userID to be used instead of user full name so that the records would be anonymous.

C. Data Transformation

For data transformation sqldf package which allows complex database queries is used in RStudio. Raw data was consisting of Time, User full name, Affected user, Event context, Component, Event name, Description, Origin, IP address. After filtering and transformation, we created a table with the fields given in Table 1.

TABLE I. ATTRIBUTES AFTER DATA TRANSFORMATION

Name	Description
UserID	ID number of the student
Visits	Total number of visits by the student
Quizzes	Number of quizzes taken by the student
Assignments	Number of submitted assignments by the student
ForumCreated	Number of forum creations by the student
ForumView	Number of forum views by the student
CourseView	Number of course views by the student
FileSubmission	Number of file submissions by the student
GradeView	Number of grade views by the student

D. Data Integration

Transformed data is integrated with the data containing course grades. Table 1 is updated by adding one more attribute "Grade". Grades are in the range of 5-10 where 5 represents failure of the student.

E. Data Exploration

The results of this step are obtained by using the sqldf package and the lattice package in R Studio which helps in querying and extracting the required data and presenting them as bar charts.

Data Exploration consists of following steps; visualizing views of lectures for each week, views of lab exercises, total visit frequency, distribution of the grades, quiz, assignment, forum, file submission, grade view frequencies and distribution of weekly visits.

F. Statistical Analysis of Data

In this step we get the summary statistics of data and correlations of visits, quizzes, assignments, forum creations, forum views, file submissions. Grade views with the course grade. We investigated the effect of these attributes on the success of the students. Correlations are found by using Pearson correlation test. Equation 1 gives the formula for Pearson correlation coefficient. R value is always between –1 and +1. r = –1 or +1 indicates a perfect linear relationship, where sign indicates the direction and r = 0 indicates no linear relationship.

$$r = \frac{\sum(x-\bar{X})(y-\bar{Y})}{\sqrt{\left[\sum(x-\bar{X})^2\right]\left[\sum(y-\bar{Y})^2\right]}} \quad (1)$$

III. Results and Discussion

A. Frequencies of student activities

Fig. 2 shows the views of the lecture slides. The first lecture which contains introduction to the course is the most visited one with more than 1200 visits; although there are 265 students registered to the course.

Fig. 3 shows the lab views. As it is the case in the course views, number of lab views decrease by time.

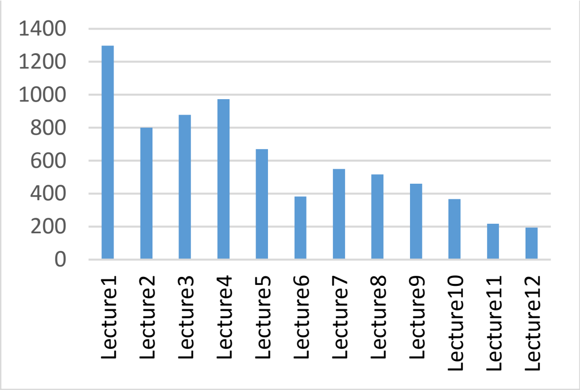


Fig. 2. Slide views per lecture

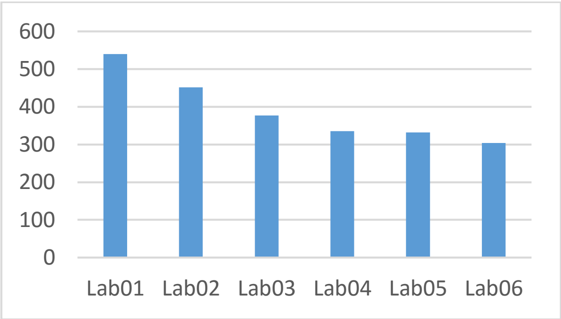


Fig. 3. Lab views

Fig. 4 shows course grades. Number of failed students is 72 out of 265. In future studies failed students' behaviours could be analysed in more details to avoid the failures.

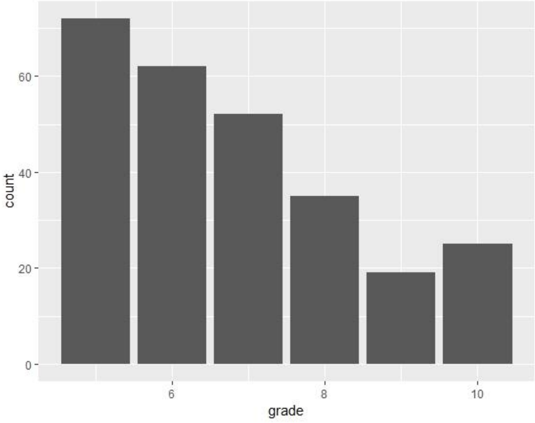


Fig. 4. 2016-2017 Grades (count represents number of students)

Table II gives the five-point summary together with the mean of each attribute of the data.

TABLE II. SUMMARY STATISTICS OF DATA

	Visits	Quizzes	Assignment	Forum Created	Forum View	Course View	File Submission	Grade View	Grade
Min	1	0	0	0	0	0	0	0	5
Q1	206.8	1	7	0	3	45.75	7	0	5
Median	281.5	2	11	0	7	65.50	11	1	6
Mean	301.9	1.938	10.87	0.1769	11.69	71.8	10.87	2.85	6.773
Q3	390.2	3	15	0	14	93.50	15.00	3	8
Max	1188	5	25	5	77	256	25.00	74	10

As it can be seen from Fig.5 The number of Moodle activities increase on the weeks which are closer to the first partial exam. There is a sharp drop out after the exam. Just before the second partial exam student activities are increasing but not as much as before the first partial exam. The possible reason for that; students who could not gain good results in the first partial exam were not interested in the learning system anymore.

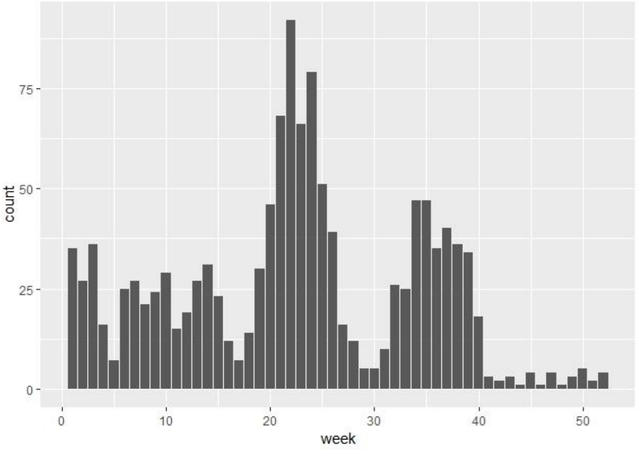


Fig. 5. Distribution of logs per week

B. Correlations

Correlations are found by using Pearson correlation test. The p-value of all attributes are less than the significance level alpha = 0.05, we can conclude that each of the selected attributes in Table I and grade are significantly correlated with the correlation coefficients given in Table II. The highest correlation is found in Assignment and File submission. The lowest correlation is found in Forum created. The reason was the maximum number of forums created by students was 5. It seems students tend to view the forums more but they are not willing to start forums very often.

TABLE III. CORRELATIONS WITH GRADE

Attributes	Correlation Coefficients
Total visits	0.55
Course view	0.43
Forum view	0.20
Forum created	0.04
Quizzes	0.38
Assignment	0.69
File submission	0.69
Grade view	0.23

Fig. 6 and Fig. 7 shows the scatter plots of correlations between visits and gained points; forum view and gained points, respectively.

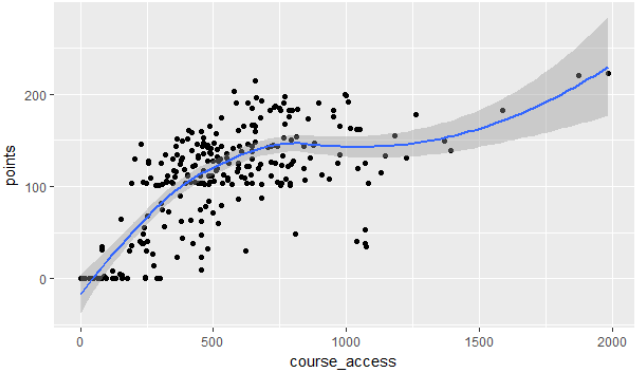


Fig. 6. Scatter Plot of Positive Correlation between total course visits and score in points

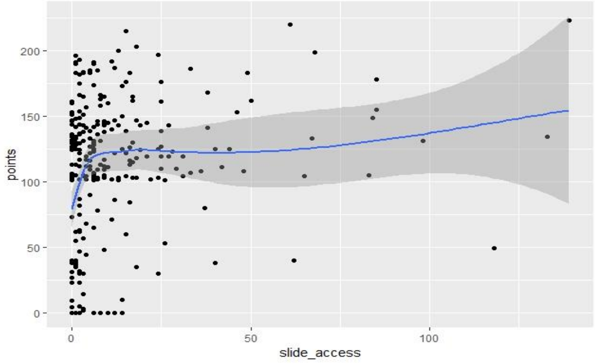


Fig. 7. Scatter Plot of Positive Correlation between forum view and score in points

IV. CONCLUSION

In this paper we used some data mining techniques such as pre-processing and statistical analysis; to discover the effect of students' online engagement and activities on their success in terms of their course grade by using R language and RStudio. The biggest advantage of using RStudio was at the steps of data pre-processing and data transformation, it allows the usage of SQL commands. In this way, data can be manipulated, filtered and transformed easily.

Present study revealed the relationship between variables which are obtained from Moodle logs and students' grade. We found positive correlations between all Moodle activities and the course grade of the students. So the more engagement brings better results in terms of success. More engaged students tend to get higher grades.

One possible application of the results for the future studies might be to develop classifiers for detecting students' engagement in the learning environment and by using prediction methods to detect students who are about to drop out. These types of methods can also be used in adaptive learning environments.

REFERENCES

[1] S. Dumais, R. Jeffries, D. M. Russell, D. Tang, and J. Teevan, "Understanding User Behavior Through Log Data and Analysis," in *Ways of Knowing in HCI*, New York, NY: Springer New York, 2014, pp. 349–372.

[2] A. Gökhan, "Profiling Students' Approaches to Learning through Moodle Logs," *Proc. Multidiscip. Acad. Conf. Educ. Teach. E-learning Prague 2015, Czech Repub. (MAC-ETeL 2015)*, no. December, p. 7, 2015.

[3] A. Konstantinidis and C. Grafton, "Using Excel Macros to Analyse Moodle Logs," *UK Res.*, no. September, pp. 4–6, 2013.

[4] Á. Figueira and Álvaro, "Mining Moodle Logs for Grade Prediction," in *Proceedings of the 5th International Conference on Technological Ecosystems for Enhancing Multiculturality - TEEM 2017*, 2017, pp. 1–8.

[5] Á. Figueira, "Mining Moodle Logs for Grade Prediction: A Methodology Walk-through," *Proc. 5th Int. Conf. Technol. Ecosyst. Enhancing Multicult.*, p. 44:1-44:8, 2017.

[6] T. Käser, N. R. Hallinen, and D. L. Schwartz, "Modeling exploration strategies to predict student performance within a learning environment and beyond," in *Proceedings of the Seventh International Learning Analytics & Knowledge Conference on - LAK '17*, 2017.

[7] M. Cocea and S. Weibelzahl, "Log file analysis for disengagement detection in e-Learning environments," *User Model. User-adapt. Interact.*, 2009.

[8] C. Romero, S. Ventura, and E. García, "Data mining in course management systems: Moodle case study and tutorial," *Comput. Educ.*, vol. 51, no. 1, pp. 368–384, 2008.

History of the GPUs

Damjan Najdov
FCSE
Ss. Cyril and Methodius University
Skopje, Macedonia
dame.najdov@gmail.com

Vladimir Zdraveski
FCSE
Ss. Cyril and Methodius University
Skopje, Macedonia
vladimir.zdraveski@finki.ukim.mk

Marjan Gusev
FCSE
Ss. Cyril and Methodius University
Skopje, Macedonia
marjan.gushev@finki.ukim.mk

***Abstract*—Graphics cards have come a long way. In the past mainly used for research, today an essential hardware component of any modern system, with an increasing set of functionalities. Graphics cards have a long history in both technological advancements and economy/market competition and popularity. This paper describes important milestones in chronological order. It focuses less on the technological advancements and more on the economical and consumer aspect of graphics cards.**

***Index Terms*—GPU, API, Graphics, History, Evolution**

I. Introduction

The definition of a graphics card is a specialized circuit to accelerate the creation of images for an output device with a screen. The term GPU (Graphics Processing Unit) was popularized by Nvidia in 1999 when they marketed their GeForce 256 as ”The world's first GPU”, and presented it as a graphics card which had transform, lightning, triangle clipping and rendering capabilities.[1] Nonetheless, graphics cards have existed since before the 1970s, and have been referred to as graphics accelerators or chips, video cards and display adapters.

The first cards were expensive and made to only work with specific hardware and software. In the 1980s discrete graphics cards started to take off. However, their non-programmable fixed-function pipeline meant that they had a fixed number of ways they could process input data and provide display output. Later, this was replaced with shaders, which are small programs that the programmer writes and tell the graphics card how to transform the data into pixel output. Eventually, the potential of their parallel nature was realized. With Cuda and OpenCL (Open Computing Language) implemented as standard general purpose GPU APIs (Application Processing Interface), GPUs could run any kind of parallelizable programs - not just generate pixel output from vertices.

II. Related work

Several articles and papers have been published summarizing the evolution of graphics cards. Chris McClanahan [17] provides a detailed, but very technical summary of the history of GPUs. Marko J. Mii in his Evolution and trends in GPU computing [18] provides a programmer's perspective to changes in the way the GPU works. The techspot article on the topic [19] is also quite technical and would be difficult to fully understand to someone not well familiar with the topic. Considering the speed at which technology evolves, the aforementioned works are a bit outdated, dating at 2010, 2012 and 2013 respectively.

III. Development Throughout the Years

A. Early Consumer Graphics (1948 - 1995)

During this period, graphics cards had very different designs, architectures and APIs, and were typically made for a specific device or computer only. Additionally, graphics cards were still very dependent on the CPU for even the most basic tasks. Before 1970s graphics cards were used mainly for research, testing or simulations. Whilrwind was a vacuum tube computer developed by MIT in 1948 that worked in parallel, could do flight simulation, and displayed on an oscilloscope.[2] In 1976 Cromemco released Dazzler, the first commercial color graphics card, which used the 8 bit S-100 bus and could output 128x128 pixels in 8 colors. It could be installed in Cromemco microcomputers only and attached to a color TV. A store owner described the reaction when he displayed a colorful pattern-generating program, the Kaleidoscope, on a color TV in his store. ”People driving by began to stop and look - they had never seen anything like it.” In a short time he caused a traffic jam and was forced to shut it down.[3]

RCA's ”Pixie” video chip (CDP1861) also released in 1976 for the RCA Studio game console, generated a monochrome 64x32 NTSC compatible signal.[4] These chips were quickly followed a year later by Atari 2600's Television Interface Adapter (TVI) which was relatively popular.

In 1978 Motorola released their MC6845 video address generator, which was the basis for IBM's Monitor Display Adapter (MDA) and Color Graphics Adapter (CGA). These were used in IBM's Personal Computer (PC) in 1981 which was widely popular. The MDA was not pixel-addressable and could only display monochrome text at 80 columns by 25 lines. Each character was drawn in a 9x14 box, giving it a total resolution of 720x350 pixels. The CGA could display:

- 640x200 in 2 colors
- 320x200 in 4 colors
- 160x100 in 16 colors

This is due to its limited framebuffer size. Game developers used a technique called dithering to overcome the very limited color palette. The MDA was more popular than the CGA, especially for business use. The author of an IBM publication said in 1981 that he had planned to purchase the CGA adapter but changed his mind after seeing how rough it is and how much sharper the MDA is, observing that ”you stare at text a whole lot more than you stare at color graphics”.[6] Motorola later that year released the MC6847 which was used in a number of first generation computers.

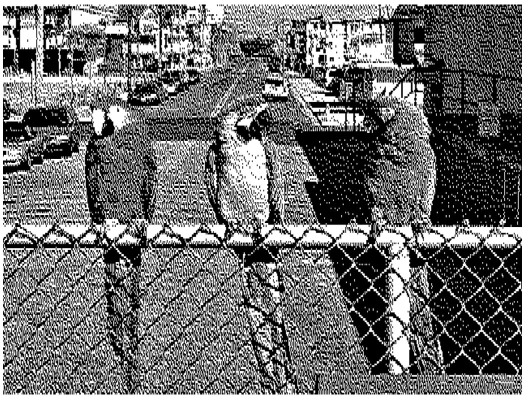


Fig. 1. 640x200, 2 colors

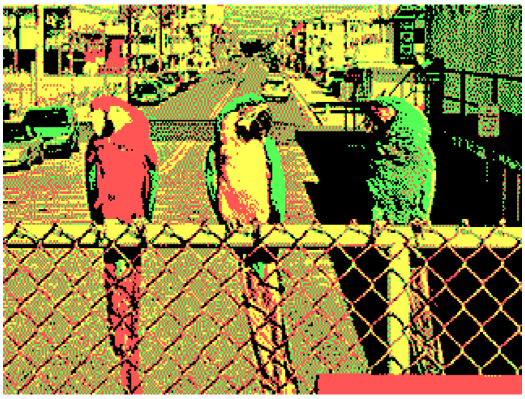


Fig. 2. 320x200, 4 colors

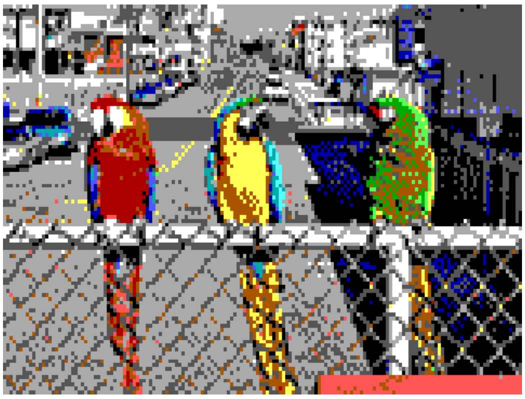


Fig. 3. 160x100, 16 colors

Commodore used a similar design in their Vic II graphics card for the Commodore 64 in 1982. It had 160x200 in 16 colors. It could reference and draw sprites and check collision detection independently of the CPU.

In 1983, on its seventh birthday Intel released the iSBX 275 Graphics controller which could display 256x256 in 8 colors or 512x512 monochrome. It had 32kb of display memory and had instructions for drawing lines, circles, rectangles and colored sprites. This was a big step forward for graphics cards.

1985 is noteworthy for the Commodore Amiga which featured three main component chips – Agnus, Denise, and Paula, that could offload a certain amount of graphics and audio calculations from the CPU. The same year, four Hong Kong Immigrants, Lee Ka Lau, Francis Lau, Benny Lau, and Kwok Yuen Ho founded Array Technology Inc. in Canada.[7] Later that year the name was changed to ATI Technologies. ATI made graphics cards for IBM and Commodore, but by 1987 was a fully independent retailer and introduced the EGA Wonder and VGA Wonder cards.[8]

In 1992 the first version of the OpenGL specifications debuted, which defines a baseline set of features and API calls that all conforming graphics cards must support. A programmer could use these calls and his code would be compatible with any card implementing OpenGL. Likewise, a CPU interfacing the card through OpenGL would be compatible with any OpenGL card. At this point discrete graphics cards which could work on various devices became more widespread.

Nvidia was founded in 1992, by three co-founders that saw the potential in graphics acceleration especially in the gaming industry.[9] In 1995 Microsoft made the DirectX specifications which would become the main competitor of OpenGL. Most high end graphics cards would implement support for both DirectX and OpenGL. ATI released the Mach 32, which provided improved bandwidth and GUI acceleration, and in 1994 the Mach 64.

B. 3D and 3dfx Interactive Voodoo (1995 - 1999)

During this period, graphics cards had become discrete and implemented standardized APIs such as OpenGL and DirectX. Many GPU focused companies emerged and started competing for the market. 3D, and the gaming industry were becoming considerably large and increasingly demanding. This was a gold mine for graphics card companies, provided that they can keep up with the increasing demands.

In 1995, Nvidia released the nv1 (or Diamond Edge 3D) graphical accelerator designed to work with Sega devices, ATI introduced the Rage 3D, Matrox the Impression card, and S3 the Virge. They could all render 2D and 3D. The nv1 was the fastest, but they were all slow, to the point where the S3 Virge was called a graphical decelerator by hardware enthusiasts.[10] The nv1 was used in the Sega Saturn game console and supported a maximum resolution of 1600x1200 with 16-bit color at 75 MHz. It used unconventional quadratic surfaces to render primitives.

In 1996, Nvidia worked on the nv2, but due to a series of disagreements with Sega, Sega took a different route and the nv2 was canceled. Ati's released the Rage 3D 2, and Matrox released the Mystique card, which was the first to allow support for DRAM expansion cards. 3dfx released the Voodoo card which was much faster than all of the competition, and soon took over 85% of the 3D market share. However, it could only output 3D and was used alongside a traditional 2D video controller. Part of the performance boost was due to its Glide API, which didn't use an abstraction layer, and the code mapped directly to Voodoo instructions. It supported up to 640x480 in 3D with bilinear filtering.

In 1997, Nvidia released the nv3 (or Riva 128), ATI released the Rage 3D Pro, Matrox released the Mystique 2 and 3dfx released Voodoo Rush. This was the first card to support what would later be called SLI, or Scalable Link Interface. This makes it possible to use two or more compatible graphics cards to work together to achieve a higher throughput. 3dfx partnered with Alliance to combine integrate 2D in the Rush and sell it as a single chip 2D/3D solution. However, it performed worse than the Voodoo due to poor integration.

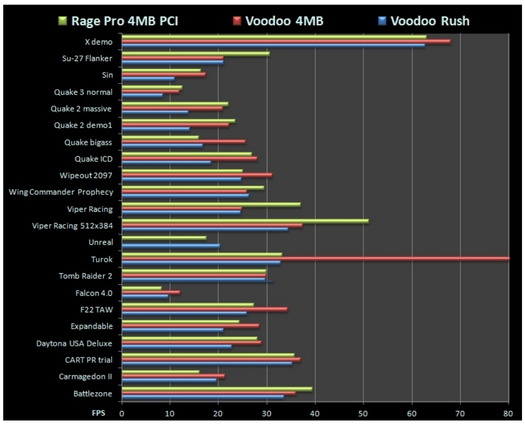


Fig. 4. Voodoo 4MB, Voodoo Rush, Rage 3D Pro Benchmarks

In 1998, 3dfx released the Voodoo2, which was fast, but it was huge and consisted of three chips each with its own memory interface, and again, was 3D only. It supported 800x600 with 16 bit color and ran at 95 MHz. [11] Competition caught up within a few months, namely the Riva TNT and Rage 128 released that year. Developers were continuously disappointed by 3dfx not delivering promised features. Later that year 3dfx released the Voodoo Banshee, but their leadership was long gone by then.

In 1999, Nvidia released the GeForce 256, which added a set of important features and was an important milestone in the history of graphics cards. It was fast, supported 3D and 2D, and offloaded transformation and lightning effects which was done on the CPU in competing products. However, it was expensive and some had a faulty analog signal which caused a blurry display. It ran at 120 MHz with 1280x1024 at most and 32 bit color output. ATI released the Rage Fury which couldn't really compete. White the GeForce 256 was ahead of its time and most games didn't utilize its new features, it paved the road for the GeForce 2, released next year, which was widely popular.

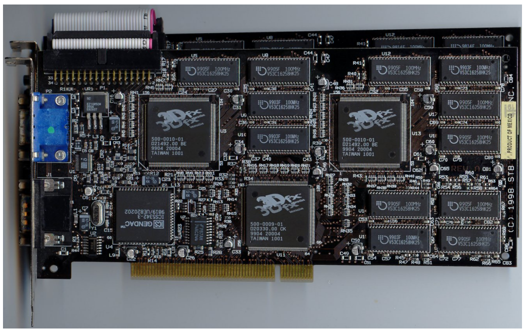


Fig. 5. Voodoo 2

C. ATI Versus Nvidia (2000 - 2006)

During this time, graphics cards started offloading more and more features from the GPU, and the OpenGL and DirectX specifications expanded and became more widespread. A lot of unsuccessful manufacturers were overwhelmed by ATI and Nvidia, and by 2001 these were the only real competitors remaining. Nvidia bought 3dfx in 2000[12], and would adopt their SLI technology in 2004. Typing the former 3dfx website (www.3dfx.com) in the address bar now redirects to Nvidia's GeForce series website.

In 2000 ATI released the first Radeon card, the R100 supporting Direct 3D 7 and OpenGL 1.3. The Rage series was no longer being supported.

In 2001 Nvidia released the GeForce 3, which featured a pixel shader allowing for much granular detail in 3D since it could produce effects on a per pixel basis.

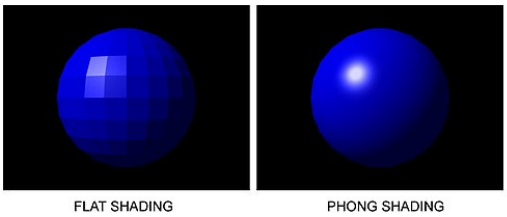


Fig. 6. Phong Shading - a technique that interpolates light across a surface. It utilizes pixel shading. The 3D model is the same.

ATI quickly added this technology in their second generation Radeon cards.

In 2002 ATI introduced the Imageon system-on-a-chip to provide graphics acceleration to mobile phones.

Over the next few years, ATI and Nvidia made gradual performance improvements with each next generation of graphics cards. In 2004, graphics cards switched from using the old AGP interface to PCIe for communication with the CPU. This year Nvidia reintroduced SLI technology in their GeForce 6. A year later ATI introduced their equivalent, the CrossFireX in the Radeon 800 series. Both technologies allowed for up to 4 graphics cards to be used in the same system to improve performance, and were very different than the old 3dfx SLI technology.

In 2006 AMD bought ATI. AMD as a CPU manufacturer wanted to get ahead of their rival Intel by offering a CPU/GPU solution on a single chip that Intel wasn't ready to offer. AMD first approached Nvidia, but Nvidia Chief Executive Jen-Hsun Huang insisted that he be the CEO of the combined company. So, AMD CEO Hector Ruiz turned to ATI instead and bought it for $5.6 billion in July, 2006. AMD had alignment and integration issues, and by the time they got back on track, Nvidia was dominating the GPU market and starting to target mobile graphics with their new Tegra architecture.

Nvidia released the GeForce 8800 GTX in 2006, which was a big, power hungry and fast graphics card. It had a large number of transistors and generated a lot of heat but became really popular for its performance.

This period also denotes the appearance of stream processing which improves the efficiency of GPU parallelism and allows for general purpose GPU computing, or GPGPU for things like cryptography, scientific simulations and any parallelizable workloads. AMD incorporated similar technology in their Radeon 2000 series in 2007.

D. Modern Days (2007 - 2019)

In this period, graphics cards have become a vital component in any computer or mobile device. AMD and Nvidia are still the primary manufacturers and Nvidia still dominates the market shares.

In 2009 AMD introduced their Eyefinity technology which allowed a graphics chip to support up to 6 displays at once. It could work in extended mode, where each display works independently, or in SLS (Single Large Surface) mode that combines the resolutions of all connected displays and tricks the operating system into believing there is a single display with the combined resolution.[14] Nvidia followed this up in 2010 calling it Surround (later Mosaic).

In 2009 a Technology company called Qualcomm bought AMD's mobile graphics division.[15] Qualcomm renamed the Imageon products to Adreno, an anagram of Radeon. AMD had said this move would cut down costs and help AMD focus on areas of profitability.

Intel started developing integrated graphics in 2010 called Intel HD Graphics. Intel sold these soldered on the same chip as the CPUs, as a single chip CPU/GPU solution. Devices that didn't have an external graphics card could still output graphics using the integrated card. It is worth noting that Intel's HD Graphics use up much less power and are much slower than dedicated GPU solutions, though performance has nearly doubled per generation.

Support for 4K and higher resolutions first came in 2012 for both Nvidia and AMD. Today, the latest Nvidia consumer flagship, the RTX 2080 Ti, supports a maximum resolution of 7680x4320 at 60 Hz.

Early 2018 marked the peak of Bitcoin and various other cryptocurrencies. Cryptocurrencies work such that if you could ”guess” a number that fits a certain cryptocurrency formula, you would be rewarded. This led to a phenomenon known as cryptocurrency mining. Graphics cards are good at this kind of problems, where the same task ought to be run with different parameters. People started massively buying graphics cards for mining, which greatly inflated the price of the graphics cards. However, it only lasted a few months and as the Bitcoin economic bubble burst the prices returned to normal.

In the September of 2018 Nvidia introduced the GeForce 20 series graphics cards, namely the RTX 2080 Ti, the RTX 2080, the RTX 2070, and a bit later in 2019 the RTX 2060. RTX cards feature special Machine Learning tensor cores and Ray Tracing, a new technology to calculate shadows in 3D environments, more similar to the way light actually works. However, these technologies saw low adoption rate, with very few games and applications utilizing Ray Tracing. Additionally, the RTX 2080 and 2080 Ti were some of the most expensive flagships ever released, with the 2080 Ti founder's edition at 891 euros. Nvidia attributes these two facts as the reason for the disappointing sales of the 2080 (Ti) and the 2070. [16] The 2060, released 5 months later, saw better sales and Nvidia expects a more profitable 2019.

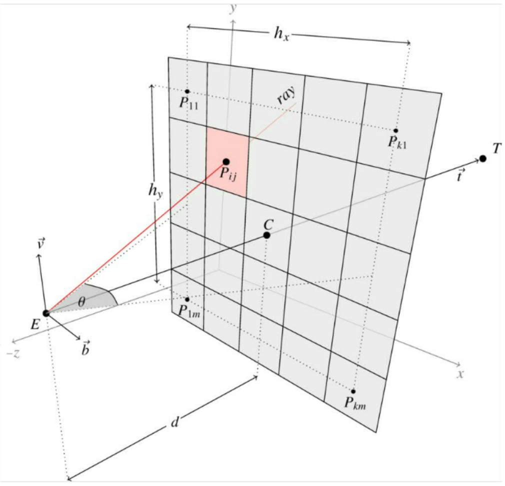


Fig. 7. Ray tracing shoots a ray through each pixel and considers what the ray intersects. Author: Kamil Kielczewski

In 2019 AMD released the Radeon 7, marketing it as the world's first 7nm GPU (hence the name). This was to compete with Nvidia's 2080 at a much lower price and a lower performance. Nvidia in an unusual naming scheme, released the GTX 1660 as a successor to the GTX 1060. Neither the Radeon 7 nor the GTX 1660 support Ray Tracing or tensor cores.

IV. Conclusion

In short, here are the most important milestones in GPU history:

1976 - Cromemco releases Dazzler, the first commercial color graphics card.

1978 - IBM releases their Monitor Display Adapter (MDA) and Color Graphics Adapter (CGA) for their IBM PC - widely popular.

1982 - Commodore releases the Vic 2 for the Commodore 64 console, which combined features from several cards.

1983 - Intel releases the iSBX 275 for their device - really fast and powerful.

1975 - Commodore Amiga is released for the Commodore. ATI is founded.

1992 - The first version of the OpenGL specifications are out, which allow any compliant graphics card to be used with any device. Nvidia is founded.

1995 - The first version of DirectX - competes with OpenGl.

1996 - 3dfx releases the Voodoo card which took over 85% of 3D market share.

1999 - Nvidia releases the GeForce 256 - added a set of important features, though then new and unsupported, paved the way for future GPUs.

2000 - ATI releases the first Radeon card - a series still in development today.

2002 - ATI introduced the Imageon System on a chip (SoC) to provide graphics acceleration to mobile devices.

2006 - AMD buys ATI. Nvidia releases the GeForce 8800 GTX - a very fast and power-hungry card which raised the bar for performance standards.

2007 - GPUs allow for stream processing, which in turn allows for general purpose GPU computing.

2009 - Qualcomm buys AMD's mobile division. AMD introduces their Eyefinity technology to allow 6 displays on the same graphics card. Nvidia follows in 2010.

2010 - Intel starts developing integrated graphics on CPUs.

2012 - Support for 4K resolutions in Nvidia and AMD cards.

2018 - The rise and fall of Bitcoin and cryptocurrency mining. Nvidia releases the RTX series which add Raytracing and Machine learning cores.

2019 - AMD releases the Radeon 7, the world's first 7nm GPU.

TABLE I
FLAGSHIP CARDS COMPARISON

GPU	Release	Transistors	Speed*
8800 GTX	2006	681M	1.00
HD 3870	2007	666M	0.94
HD 4870	2008	956M	1.71
9800 GTX	2008	754M	1.00
GTX 280	2008	1.4B	1.57
HD 5870	2009	2.15B	3.42
GTX 480	2010	3B	5.70
GTX 580	2010	3B	6.55
HD 6970	2010	2.64B	4.60
GTX 680	2012	3.5B	7.44
HD 7970	2012	4.3B	6.87
GTX 780 Ti	2013	7B	11.65
R9 290X	2013	6.2B	9.69
GTX 980	2014	5.2B	12.57
R9 390X	2015	6.2B	11.20
GTX 980 Ti	2015	8B	14.86
GTX 1080	2016	7.2B	16.10
RX 480	2016	5.7B	10.64
GTX 1080 Ti	2017	12B	18.40
RX Vega 64	2017	12.5B	15.20
Titan V	2017	21.1B	18.71**
RTX 2080 Ti	2018	18.6	21.90
Radeon VII	2019	13.2B	17.8

* Average relative performance for various games and tasks.[13]
** The Volta architecture on the Titan V is optimized for neural networks and deep learning tasks, which are not included in the benchmarks.

REFERENCES

[1] "NVIDIA Launches the World's First Graphics Processing Unit: GeForce 256". Nvidia. 31 August 1999.
[2] Redmond, Kent C.; Smith, Thomas M. Project Whirlwind: The History of a Pioneer Computer. Bedford, MA: Digital Press. ISBN 0-932376-09-6. 1980.
[3] Veit, Stan (March 1990). "Cromemco - Innovation and Reliability". Computer Shopper. 3. 10 (122): 481-487.
[4] Popular Electronics Magazine, July 1977
[5] Curran, Lawrence J.; Shuford, Richard S. November 1983. "IBM's Estridge". BYTE. pp. 88-97
[6] Dievendorff, Dick (1981). IBM Personal Computer Questions and Answers. IBM. p. 25
[7] "The Information Technology 100: 90: ATI Technologies". BusinessWeek. BusinessWeek. 2005
[b8] istory of AMD, amd.com, archived October 12, 2007 at the Wayback Machine
[b9] S3 Graphics", Vintage3d.org
[8] Nusca, Andrew (16 November 2017). "This Man Is Leading an AI Revolution in Silicon Valley-And He's Just Getting Started". Fortune. Archived from the original on 16 November 2017. Retrieved 28 November 2017.
[9] "Rush(ed?)" 3dfx Voodoo Rush Review http://vintage3d.org/3dfx2.php
[10] "To our valued customers" 2000, 3dfx website http://3dfx.com
[11] Passmark Benchmarks, https://www.videocardbenchmark.net/GPU_mega_page.htm
[12] AMD support, https://support.amd.com/en-us/search/faq/13
[13] Ferguson, Scott (January 20, 2009). "AMD Sells Handset Division to Qualcomm for $65 million". eWeek. Retrieved June 6, 2014.
[14] Nvidia explains disappointing RTX 2080 / 2070 sales, notebookcheck, Bogdan Solca 15.02.2019
[15] https://pdfs.semanticscholar.org/2479/80e834f1c8f684d85067402f950930e6af91.pdf
[16] https://ieeexplore.ieee.org/document/6240658/authors
[17] https://www.techspot.com/article/650-history-of-the-gpu/

Human activity recognition using sensor recordings from passive infrared and microwave radar sensors

Aleksandra Bozhinoska
Ss. Cyril and Methodius University in Skopje
Faculty of Computer Science and Engineering
Skopje, Republic of North Macedonia
aleksandra.bozinoska@students.finki.ukim.mk

Dejan Gjorgjevikj
Ss. Cyril and Methodius University in Skopje
Faculty of Computer Science and Engineering
Skopje, Republic of North Macedonia
dejan.gjorgjevikj@finki.ukim.mk

Abstract*—Human activity recognition is a challenging and wide-spread research topic among the data scientists. It has ever-increasing importance in human monitoring approaches used mainly in Ambient Assisted Living (AAL) and healthcare. The most demanding task regarding this field of research is achieving high recognition accuracy while maintaining low equipment costs by using fewer simple and inexpensive sensors. An experiment has been conducted by installing three identical data collection modules in a controlled environment for human activity recording. The data collection modules were composed of Arduino microcontroller and two modified low-cost ambient sensors: microwave radar sensor and passive infrared (PIR) sensor. The sensors were modified to obtain their analog outputs that were logged on a SD card. The experiment was conducted on six subjects performing seven different activities. This has produced almost 2.5 hours of recorded sensor measurements that were then labeled with the corresponding activity producing data set of 654061 entries. Different approaches for feature extraction and preliminary tests for activity recognition were conducted. Different sliding window sizes, as well as considering only certain sensors (of the six possible) and their combinations were also examined.

***Index Terms*—human activity recognition; passive infrared sensor; microwave radar sensor; Arduino; sliding window technique**

I. Introduction

The number of elderly people is rapidly growing as a proportion of the total population in most developed countries around the world [1]. Significant number of them live alone within their own house. To be able to function independently at home, individuals have to be able to perform Activities of Daily Living (ADLs) [2] such as eating, dressing up, cooking, drinking, and taking medicine. Automating the recognition of the activities is an important step towards monitoring the functional health of a smart home resident.

Advancements in sensor technology, the declining costs of sensors and their vast availability opened up unprecedented opportunities for a wide variety of industrial, scientific, commercial, agricultural and military applications, such as home automation, health care, emergency response, smart transportation, infrastructure protection, and others [3]. Pervasive sensing technologies are becoming more and more popular offering new opportunities in smart homes such as providing health monitoring and assistance to individuals experiencing difficulties living independently at home. In order for the functional health of smart home residents to be monitored the system should recognize and track the activities that people perform at home.

Human activity recognition has attracted a lot of research activity in several fields like ambient assisted living, sports injury detection, elderly care, rehabilitation, and entertainment and surveillance in smart home environments. Most of the research was conducted using data collected by wearable sensors [4][5]. As wearable sensors smartphones, smart bands, and dedicated sensor nodes have been used. Approaches that are using unobtrusive ambient sensors were also reported [6]. Among the unobtrusive sensors use of sensor nodes equipped with acoustic sensors [7], ultrasound sensors [8], pyroelectric motion detection sensors [9] and recently microwave radar sensors [10] has been reported for human activity recognition. They are considered as essential for number of enabling technologies for independent living by the elderly such as the ambient assisted living systems (AALS).

In this paper, we present initial study of human activity recognition and identification using very cheap passive infrared (PIR) and microwave radar sensors. Three data collection modules developed around the sensors were placed in a room where volunteers were instructed to perform seven different activities in random order. The sensor measurements during the experiment were recorded by the modules and then labeled with the corresponding performed activity. Initial experiments considering different approaches for feature extraction and machine learning for automatic activity recognition were performed and the obtained results are presented and discussed.

The remainder of this paper is organized as follows: In section 2 conducting of the experiment is documented and section 3 explains the process of labeling of the gathered data. Section 4 elaborates the suggested feature extraction techniques and section 5 documents the examined machine learning approaches. Section 6 discusses and compares the used machine learning approaches and finally section 7 summarizes the results of the documented work and suggests future plans for improvement.

II. Conducting the experiment

For the needs of human activity recognition experiment, a controlled environment was created for the realization of the following seven activities: sitting on the bed, sitting at the table, walking out of the room, walking into the room, walking around the room, eating at the table and laying on the bed. For that purpose, an isolated room was prepared and supplied with only the needed furniture elements. The floorplan of the prepared room for activity recording is given in Fig.1. Three identical data collection modules were placed on three distant locations in the room, namely the two side walls and the ceiling. The modules consisted of Arduino microcontroller board, real-time clock (RTC) microSD card adapter and two motion detection sensors: passive infrared sensor HC-SR501 (PIR) and microwave radar sensor RCWL-0516 [11]. The sensors were modified to obtain the analog signal that was digitized to 10-bit precision and logged on the SD card at rate of 25 samples per second. The clocks of the sensors were synchronized before the experiment to guarantee alignment in the sensor measurements taken by the separate data collection modules.

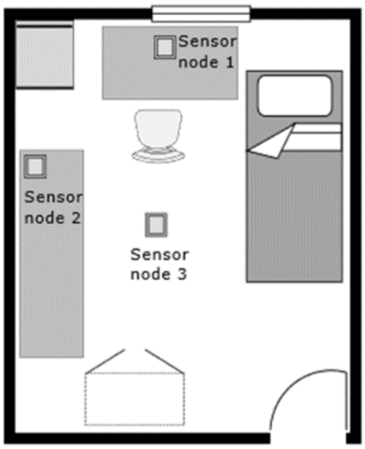


Fig. 1. Floorplan of the room where the sensors were placed

Six volunteers were instructed to perform the activities in the room one by one while the data streams from the 3 pairs of sensors were recorded. Each of the subjects was instructed on the next activity to perform by continuous delivery of messages to his mobile phone using previously developed simple mobile application. The activities were provided in random order taking into consideration the same activity not to be repeated twice in a row. The subjects were also continuously filmed by a dashboard camera for the purpose of ground truth labeling of the data. The experiment produced about two and a half hours of sensor recordings that after the cleansing resulted in 654061 entries.

The recording was performed on micro SD card in each of the modules, in digital compressed format that contains precise timestamp of each sample and also readings from some other sensors (ambient temperature, lighting and noise level) that were not used in this experiment. The data from the three modules was combined in time aligned fashion and only the readings from the PIR and the microwave radar sensors from each of the modules were considered.

III. Data fusion and labeling

The ground truth labeling was performed manually by following the activities of the subjects in the filmed videos. Nine different labels were used for labeling the entire entry set of sensor recordings, namely the labels were: SB (sitting on the bed), ST (sitting at the table), WO (walking out of the room), WI (walking into the room), WA (walking around the room), ET (eating at the table), LB (laying on the bed), SD (standing in one place) and one more used to denote subject absence or empty room, denoted as no activity - (NA). Although the last two were not part of the initial set of activities, they were added after being recognized as a potential source of confusion by the classifiers in the phase of activity recognition. The labeled experiment data consisted of 553 non-overlapping consecutive activities undertaken by the subjects with average duration time of 15.89 seconds. Statistics about the label distribution are presented in Table 1.

Table 1. Class distributions among the labeled samples

Target class	Number of samples	Percent of total number of samples
SB	31271	14,34
ST	28032	12,86
WO	4046	1,86
WI	3846	1,76
WA	37587	17,24
ET	17831	8,18
LB	30870	14,16
SD	13535	6,21
NA	51002	23,39

IV. Data processing and feature extraction approaches

A sliding window technique was exploited in the feature extraction approaches in order to capture recent history context of measured values for each of the sensors. By experimenting with different sliding window lengths varying from 2 to 20 seconds, the optimal values of 3, 4 and 5 were identified. In all the feature extraction approaches, new generated data sets were produced by applying computations on the series of values for the individual sensors in the current sliding window and then combining the new features. The feature extraction approaches were more efficient when applied to the combined data set of entries from all three modules, generated by joining the corresponding entries from each module in one entry (corresponding are the entries with the corresponding serial numbers). The following grouping operations were applied to the series of values for the 6 different sensors in the combined data set as different feature extraction approaches:

A. Averaging (computing cumulative average, weighted moving average and exponential weighted moving average)

Three different averaging approaches were examined as possible feature extraction techniques. First evaluated averaging approach was the cumulative average, or computing the standard average of all sensor measurements in the current window for each sensor. This approach gives equal importance to all the sampled values in the previous n-seconds. The number of entries in the sliding window was computed as $n * 25 + 1$ (with n being the window size in seconds), considering the sampling rate. The complete equation is defined as:

$$CMA_{n,si} = \frac{v_{1,si}+ v_{2,si}+\cdots+ v_{k,si}}{k} \tag{1}$$

where $k = n * 25 + 1$ and $1 \le i \le 6$, with si being the i-th sensor and n the sliding window size in seconds.

Two more averaging techniques were utilized in order to impose higher importance to the most recently sampled values while computing the averaged sensor values. Therefore, weighted moving average technique was applied to the values in the window, defining weights as consecutive integers in the interval [1, number of values in the window] and assigning them to the ordered sensor values. The complete equation for the weighted moving average is defined as:

$$WMA_{n,si} = \frac{w_1 * v_{1,si}+ w_2 * v_{2,si}+\cdots+ w_k * v_{k,si}}{w_1+w_2+\cdots+w_k} \tag{2}$$

where $w_j = j,\ k = n * 25 + 1$ and $1 \leq i \leq 6$, with si being the i-th sensor and n the sliding window size in seconds.

The last considered averaging technique was exponential weighted moving average with weights decreasing exponentially from latest to the earliest sampled sensor value in the window. The computation formula is defined as:

$$EWMA_{n,si} = \frac{w_k * v_{1,si} + w_{k-1} * v_{2,si} + \cdots + w_1 * v_{k,si}}{w_1 + w_2 + \cdots + w_k} \qquad (3)$$

where $w_j = (1 - \propto)^{j-1}, \propto = \frac{2}{k+1}, k = n * 25 + 1$ and $1 \leq i \leq 6$, with si being the i-th sensor and n the sliding window size in seconds. Fig. 2 visualizes the different rates of weight decrease in weighted moving average and exponential weighted moving average approaches.

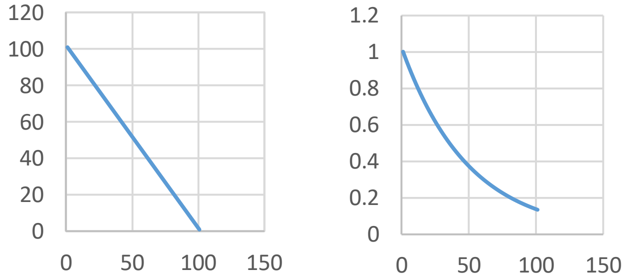


Fig. 2. Decrease of weights in WMA approach (left) and EWMA approach (right) for assumed 4 seconds sliding window size

B. Computing number of zero crossings

One proposed feature for computing the rate of change of the sensor values in the window was modified zero crossings approach. Namely this feature represented the count of occurrences when two consecutive measured values in the sequence appeared in different subintervals of the interval of all possible values [0-1023]. The first subinterval is the interval containing values less than 512 and the second one is the interval containing greater values that 512.

C. Computing number of local extremes

Another examined feature extraction approach was computing the number of local extremes considering the discrete values in the sliding window for each of the sensors. The feature was extracted by first computing the slope of the function for each pair of consecutive discrete values in the sliding window and then counting the number of pairs of consecutive slopes in the computed sequence with opposite signs. The used computation formula is defined as:

$$slope_{j,si} = \frac{v_{j,si} - v_{j-1,si}}{t_j - t_{j-1}} \qquad (4)$$

where $2 \leq j \leq k, k = n * 25 + 1$ and $1 \leq i \leq 6$, with si being the i-th sensor and n the sliding window size in seconds.

D. Integrating

Integration was used as a feature extraction approach in order to capture the magnitude of measured values for each of the sensors in the sliding window. Integrals were computed on the function represented by pairs of transformed measured sensor value and time of measurement. Measured values were transformed by first subtracting 512 (the median from all possible digitized values) and then computing absolute value. Time of measurement in the interval was represented as sequence of discrete values from 0 to 40 * window-size- in-entries with step of 40 (because of sampling rates of the values). Simpson's rule for approximating integral of function represented by discrete pairs of values was used by calling its implementation in scipy.integrate library in Python [12].

V. Experiments and Results

Different data sets were produced by applying one or more of the proposed feature extraction approaches in the previous chapter to the combined data set of all six sensors. Each of the approaches was used to compute the target feature for each of the 6 sensors so the number of the features in the generated data set was number-of-combined-approaches * 6. Two alternatives for the target label of the computed features were tested, namely taking the label of the last test entry or taking the label of the middle test entry in the current window as the target class label. The experiments regarding generating the data sets and then applying machine learning algorithms were conducted using Jupyter Notebook with Python 3.7.0 and scikit-learn library version 0.19.2. The models were tested on three different classifiers: Random forest classifier, Artificial neural network and Support vector machine. The random forest classifier included 500 estimators, ANN consisted of two hidden layers of 200 and 100 neurons, and the kernel of SVM classifier was Radial Basis Function with $\gamma = \frac{1}{number\ of\ features}$ and $C = 1.0$. Overlapping and non-overlapping sliding windows were examined as well as different sliding window sizes. The overlapping window approach showed to be troublesome for proper testing because of the unequal class label distribution over the data set and also sequential dependency of the consecutive entries (in order to produce valid values for each of the features). The non-overlapping approach on the other side, was trained on the 80% of the samples in the data set and then evaluated on the remaining 20% (considering equal target class distributions after shuffling all the samples). It was examined on the data sets generated by applying only the individual feature extraction approaches as well as the paired averaging techniques (cumulative average, weighted moving average and exponential weighted moving average) with zero crossings, local extremes and integrals. Two more combinations of three different feature extraction approaches were also examined, namely: Cumulative average + Local extremes + Integrals and Exponential weighted moving average + Local extremes + Integrals. The evaluation results are presented in Table 2 and Table 3. Table 2 summarizes the recognition accuracy by feature extraction approach and by classifier using the last test label variant and Table 3 presents the same information for the middle test label approach.

VI. Discussion

From the provided recognition accuracies in Table 2 And Table 3 we can conclude that Random forest classifier generally provided the highest recognition accuracies compared to the other two classifiers, although all the results produced by all classifiers are pretty close and similar. The

Table 2. Recognition accuracy by feature extraction approach and by classifier using the last test label approach given in percent

Classifier	Random Forest		ANN		SVM	
Window size	3s	4s	3s	4s	3s	4s
Cumulative average (CA)	21,95	19,44	15,51	20,37	24,22	21,06
Weighted moving average (WMA)	23,00	20,83	17,60	14,81	23,00	25,00
Exponential weighted moving average (EWMA)	22,82	20,37	18,12	15,97	20,73	24,54
Zero crossings	23,00	24,31	24,74	24,31	25,44	25,46
Local extremes	24,74	24,31	24,04	23,61	25,61	23,84
Integrals	22,47	25,93	16,72	15,28	21,43	22,92
CA + Zero crossings	24,74	25,93	15,85	19,44	24,74	23,38
CA + Local extremes	25,78	24,54	15,68	13,89	22,30	24,07
CA + Integrals	25,09	23,84	19,34	20,83	24,39	23,38
WMA + Zero crossings	26,13	29,17	23,34	18,98	24,39	23,61
WMA + Local extremes	27,00	25,46	15,67	20,37	24,39	26,16
WMA + Integrals	25,96	26,16	16,72	14,35	21,95	22,92
EWMA + Zero crossings	26,83	23,84	19,86	18,75	24,56	27,78
EWMA + Local extremes	25,44	23,38	13,24	18,98	22,65	24,54
EWMA + Integrals	25,26	25,69	15,33	22,22	24,91	22,22
CA + Local extremes + Integrals	26,48	28,94	17,60	20,37	23,00	23,61
EWMA + Local extremes + Integrals	27,70	25,46	20,91	17,82	24,22	21,30

Table 3. Recognition accuracy by feature extraction approach and by classifier using the middle test label approach given in percent

Classifier	Random Forest		ANN		SVM	
Window size	3s	4s	3s	4s	3s	4s
Cumulative average (CA)	23,17	19,68	19,34	14,58	22,65	21,06
Weighted moving average (WMA)	20,38	22,22	16,20	16,90	22,82	22,45
Exponential weighted moving average (EWMA)	18,99	20,60	10,98	16,20	22,30	21,30
Zero crossings	21,78	22,92	23,87	22,92	24,39	25,93
Local extremes	22,47	21,30	21,78	21,53	24,56	21,76
Integrals	23,34	22,22	21,25	12,27	23,00	22,92
CA + Zero crossings	23,52	28,47	19,51	15,51	22,65	26,16
CA + Local extremes	26,31	26,62	14,29	17,82	24,22	22,69
CA + Integrals	22,47	28,94	13,76	20,83	21,25	25,23
WMA + Zero crossings	26,66	27,78	15,33	22,92	26,31	24,07
WMA + Local extremes	24,39	26,62	22,13	16,44	23,52	25,00
WMA + Integrals	27,18	28,24	14,29	15,51	24,39	25,23
EWMA + Zero crossings	24,74	26,16	18,12	11,34	22,65	23,38
EWMA + Local extremes	24,22	23,61	21,08	11,81	23,17	23,15
EWMA + Integrals	26,48	25,46	21,60	17,36	26,13	23,61
CA + Local extremes + Integrals	28,22	32,00	20,91	16,67	25,78	24,77
EWMA + Local extremes + Integrals	26,31	29,86	16,20	15,05	24,56	23,84

best recognition accuracies were achieved with 4-second-sized window and using the middle test label approach. Because of the very limited number of subjects and therefore number of entries in the data set, the highest individual sample recognition accuracy was 32% using the combination of cumulative average, local extremes and integrals as feature extraction techniques. Probably the results produced by ANN and SVM classifiers could be improved by fitting the classifier characteristics and also the overall recognition score by enlarging the testing data set.

VII. Conclusion

In this work, various approaches to feature extraction for human activity recognition on a custom data set were presented. The data set was created over captured sensor measurements in controlled environment in which six volunteers were performing seven different activities. Different feature extraction techniques using sliding window approach, as well as different combination of features and considered sensors were investigated. Numbers of experiments of automatic activity recognition using several classifiers were conducted. Considering the very limited data set and the variety of activities and the way they can be performed, an individual entry classification rate of 32% was obtained. Collecting a bigger data set and considering alternative feature extraction and feature fusion approaches are planned for further research.

ACKNOWLEDGMENT

This research was partially funded by the Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje.

REFERENCES

[1] "World Population Ageing 2015", (ST/ESA/SER.A/390), United Nations, Department of Economic and Social Affairs, Population Division (2015).

[2] V. Wadley, O. Okonkwo, M. Crowe and L.A. Ross-Meadows, "Mild Cognitive Impairment and everyday function: Evidence of reduced speed in performing instrumental activities of daily living", American Journal of Geriatric Psychiatry 16 (2007), 416424.

[3] O. Kanoun and H. Trankler, "Sensor Technology Advances and Future Trends", IEEE Transactions on Instrumentation and Measurement, vol. 53, no. 6, pp. 1497-1501, 2004.

[4] O. Lara and M. Labrador, "A Survey on Human Activity Recognition using Wearable Sensors", IEEE Communications Surveys & Tutorials, vol. 15, no. 3, pp. 1192-1209, 2013.

[5] F. Attal, S. Mohammed, M. Dedabrishvili, F. Chamroukhi, L. Oukhellou and Y. Amirat, "Physical Human Activity Recognition Using Wearable Sensors", Sensors, vol. 15, no. 12, pp. 31314-31338, 2015.

[6] E. Salomons, P. Havinga and H. van Leeuwen, "Inferring Human Activity Recognition with Ambient Sound on Wireless Sensor Nodes", Sensors, vol. 16, no. 10, p. 1586, 2016.

[7] J. Sim, Y. Lee and O. Kwon, "Acoustic Sensor Based Recognition of Human Activity in Everyday Life for Smart Home Services", International Journal of Distributed Sensor Networks, vol. 11, no. 9, p. 679123, 2015.

[8] J. Sim, Y. Lee and O. Kwon, "Acoustic Sensor Based Recognition of Human Activity in Everyday Life for Smart Home Services", International Journal of Distributed Sensor Networks, vol. 11, no. 9, p. 679123, 2015.

[9] X. Luo, Q. Guan, H. Tan, L. Gao, Z. Wang and X. Luo, "Simultaneous Indoor Tracking and Activity Recognition Using Pyroelectric Infrared Sensors", Sensors, vol. 17, no. 8, p. 1738, 2017.

[10] G. Diraco, A. Leone and P. Siciliano, "A Radar-Based Smart Sensor for Unobtrusive Elderly Monitoring in Ambient Assisted Living Applica- tions", Biosensors, vol. 7, no. 4, p. 55, 2017.

[11] D. Gjorgjevikj, Gj. Madjarov, "Data Collection Module for Human Activity Recognition", in Miroslav Kotevski (edt.), Proceedings of 15th International Conference, ETAI 2018, pp. ETAI1-1, Ohrid, Republic of North Macedonia, 20-23 September 2018.

[12] Scipy.integrate v1.1.0, https://docs.scipy.org/doc/scipy-1.1.0/reference/index.htm.

Implementation of novel faculty e-services for workflow automatization

Dimitar Kitanovski, Aleksandar Stojmenski, Kostadin Mishev,
Ivan Chorbev, Vesna Dimitrova
”Ss. Cyril and Methodius” University in Skopje
Faculty of Computer Science and Engineering
”Rugjer Boshkovikj” 16, 1000 Skopje, Republic of Macedonia

***Abstract*—This paper presents a brief overview of the concepts for collaboration between various systems developed for the Faculty of Computer Science and Engineering in Skopje. Web technologies such as the HTTP, originally designed for human-to-machine communication, is utilized for machine-to-machine communication, more specifically for transferring machine-readable data in web service formats such as JSON. By using this kind of web technology and communication we can create various software applications suitable for various needs.This kind of web based software applications enable automatization and drastically eased and accelerated the entire procedure whose initial steps in the past was manually.**

This paper gives a brief overview of two novel systems which are integrated in the faculty software architecture. Software's for master thesis submission and student surveys are integrated as a part of the core systems. The system's network is collaborating using web services, central authentication services and data sharing which is based on cross-platform interfaces.

Keywords: faculty systems; collaboration; systems integration; web services; administration; cross-platform

I. Introduction

Workflows automatization is a complex process that integrates process automation tools to replace manual and paper-based processes. Workflow Automation refers to the design, execution, and automation of processes based on workflow rules where human tasks, data or files are routed between people or systems based on pre-defined business rules. This paper describes two software applications that are used to improve the work processes of the faculty. In order to develop this kind of software applications that are easy to integrate into existing system architecture and maintenance, novel software design practices are required. For this purpose, different design patterns are used to facilitate the communication between systems and integration to the current system architecture.

Different problems and their possible solutions are presented, regarding systems lifecycle, architecture, process, interface, synchronization and security. Each application endpoint exports OAuth2 security protocol functionalities for system authentication and authorization. Following the principles of the OAuth2 protocol, each server authenticates the users using bearer tokens. Furthermore, the communication protocol adopts the JSON data format as a primary exchanging throughput over a HTTP communication channel. [1]

II. Background work

Although a lot of work and progress has already been done in the area of web services in the past years, efforts have been mostly focused on service description models and languages, and on automated service discovery and composition[2]. The term Web services is used frequently nowadays, although sometimes it is very ambiguous. Existing definitions of the terms vary from generic to specific and restrictive. One definition is that a Web service is seen as an application accessible to other applications over the Web [3]. This is a very open definition meaning that anything with a URL address is a Web service. It can include a CGI script or refer to a program accessible over the Web with a stable API, published with additional descriptive information on some service directory. A more precise definition is provided by the UDDI consortium, which characterizes Web services as “self-contained, modular business applications that have open, Internet-oriented, standards-based interfaces” [4]. This definition is more detailed, placing the emphasis on the need for being compliant with Internet standards. In addition, it requires the service to be open, which essentially means that it has a published interface that can be invoked across the Internet. In spite of this clarification, the definition is still not precise enough. For instance, it is not clear what it is meant by a modular, self contained business application. A step further in refining the definition of Web services is the one provided by the World Wide Web consortium (W3C), and specifically the group involved in the Web Service Activity: “a software application identified by a URI, whose interfaces and bindings are capable of being defined, described, and discovered as XML artifacts. A Web service supports direct interactions with other software agents using XML-based messages exchanged via Internet-based protocols”. The W3C definition is quite accurate and also hints at how Web services should work. The definition stresses that Web services should be capable of being “defined, described, and discovered,” thereby clarifying the meaning of “accessible” and making more concrete the notion of “Internet-oriented, standards-based interfaces.” [5] [6] It also states that Web services should be “services” similar to those in conventional middleware. Not only they should be “up and running,” but they should be described and advertised so that it is possible to write clients that bind and interact with them. In other words, Web services are components that can be integrated into more complex distributed applications. The W3C also states that XML is part of the solution. Indeed, XML is so popular and widely used today that, just like HTTP and Web servers, it can be considered as being part of Web technology. There is little doubt that XML will be the data format used for many Web-based interactions. Note that even more specific definitions exist. For example, in the online technical dictionary Webopedia, a Web service is defined as "a standardized way of integrating Web-based applications using the XML, SOAP, WSDL, and UDDI open standards over an Internet protocol backbone. XML is used to tag the data, SOAP is used to transfer the data, WSDL is used for describing the services available, and UDDI is used for listing what services are available" [7]. Specific standards that could be used for performing binding and for interacting with a Web service are mentioned here. These are the leading standards today in Web services. As a matter of fact, many applications that are "made accessible to other applications" do so through SOAP, WSDL, UDDI, and other Web standards. However, these standards do not constitute the essence of Web services technology: the problems underlying Web services are the same regardless of the standards used. This is why, keeping the above observations in mind, we can adopt the W3C definition and proceed toward detailing what Web services really are and what they imply.

Web services were developed as a solution to (or at least as a simplification of) the system integration problem[8]. The main benefit they bring is that of standardization, in terms of data format (JSON), interface definition language (WSDL), transport mechanism (SOAP) and many other interoperability aspects. Standardization reduces heterogeneity and makes it therefore easier to develop business logic that integrates different (Web service-based) applications. Web services also represent the most promising technologies for the realization of service-oriented architectures (SOAs), not only within, but also outside companies' boundaries, as they are designed to enable loosely-coupled, distributed interaction [9]. While standardization makes interoperability easier, it does not remove the need for design patterns that include adapters and mediators. Different Web services may still support different interfaces and protocols. For example, although two map or driving direction services may support JSON or XML and use SOAP over HTTP as transport mechanism, they may still provide operations that have different names, different parameters, and different business logic or protocols. In addition, other opportunities enabled by Web services have an implication in terms of adaptation needs. In fact, having loosely-coupled and B2B interactions imply that services are not designed having interoperability with a particular client in mind (as it was often the case with CORBA-style integration) [10]. They are designed to be open and possibly without knowledge, at development time, about the type and number of clients that will access them, which can be very large. The possible interactions that a Web service can support are specified at design time, using what is called a business protocol or conversation protocol. A business protocol specifies message exchange sequences that are supported by the service, for example expressed in terms of constraints on the order in which service operations should be invoked. Another studied solution is to make system integration with ActiveXML which utilities peer-to-peer interaction between nodes and specifies special data design and ActiveXML web services[11].

III. Systems architecture

The Faculty of Computer Science and Engineering continues the development of e-platform for student and staff services by providing new e-services and their adaptation with machine interfaces to the central data repository. Such services provide simplification and acceleration of the Faculty administrative workflows by providing easy-to-use interfaces avoiding congestion and bottle-neck scenarios. The system architecture that is discussed in this paper consists of several different subsystems which work as a part of the architecture provided in [12]. That means that the core of the subsystems is a common part which ties the entities as soft links providing scalable and reusable patterns for the purpose of interoperable services.

A. Architecture Core

The core of the service architecture is implemented in Microsoft .NET MVC technology. The authentication process is handled by the Central Authentication Service (CAS) which is implemented in Java. The CAS service involves a back-end service, that does not have its own HTTP interface, but communicates with a web application. The Service manager implements a protocol which is platform independent (JSON based). All applications and services in the system are communicating and synchronizing using this protocol. The service manager is implemented in C Web Api and is used as a mediator for control messages exchange, storing permission access rules, identifying the status of the services (running, failed, blocked. . .) and enabling intra service communication. As a result of successful authentication, the user obtains JWT token which is passed as authentication header, providing stateless communication between the client browser and the server. The JWT token is used as a key reference for the user credentials. Users identity management is handled by Active Directory. Active Directory is interconnected with the CAS service and serves information about the user credentials. CAS service queries the AD to enable user single sign on authentication for multiple third-party services. User authorization i.e. the role of each user for specific service is handled by each service individually. That means that each services implements its own many-to-many relationship which stores the information about the grant tickets associated to each user in the application. If the user does not contain any grant to the application, he will not be able to access it. The authorization is handled immediately after successful authentication to the CAS service. It is provided authentication by the user group. That means that is the user is a part of the group students, he will obtain different grant that the user from the group professors. This properties can be overridden by specifying the grant for each user individually. The grant with higher weight i.e. with stronger permissions wins the authorization process. The intercommunication among this services is realized with REST JSON-based web services by using the ASP.NET Web API Framework. Such core enables development of software applications that will facilitate the workflow among students, administrative and teaching staff in the faculty with implementation of several use-case scenarios. The current system architecture contained systems for student request service, consultation management service, absence workflow automatization, diploma thesis submission.... Two new systems are modulary added to the architecture, namely systems for master studies submission and student surveys.

B. *Master thesis submission*

This system is developed in order to facilitate the whole process of master thesis submission, approval and status tracking. Having in mind that the problem is complex, the system itself contains number of workflows in order to cover all the possible scenarios.

In order to implement the process of master thesis submission, we identified the following steps in forementioned workflow:

- The first step of the master thesis submission process is actually a set of several sub-processes conducted by the student and his supervisor. First of all, the student chooses a supervisor and delivers documents which are necessary for the master thesis application. Then the supervisor approves the proposed documents and assigns a committee for the particular master thesis.
- Then the documents are being verified by the faculty student affairs, the secretary and the vice-dean for academic affairs. If some of the documents is missing or is not in the appropriate format, it can be resubmitted by the student, taking him few steps back. In every step, all of the involved roles have access to the documents which are uploaded or changed in the system.
- Next is uploading the draft version of the thesis and its validation by the mentor and the faculty administration. If the student hasn't uploaded a draft version within a year, the attaching master thesis status automatically closes and the whole process is pushed back to the beginning. Otherwize, the validation takes place from the mentor,members of the committee, the secretary of faculty and the member of Teaching Scientific Commission. In this chain of validation, if something is not valid, the student is obliged to make the requested changes in the proposal.
- The last step of the master thesis submission process is submitting the final text of the thesis and determining a date for public defence. In this step, the committee members, can create notes for the completeness of the thesis. After the comments, the secretary approves again and the public defence is being scheduled.

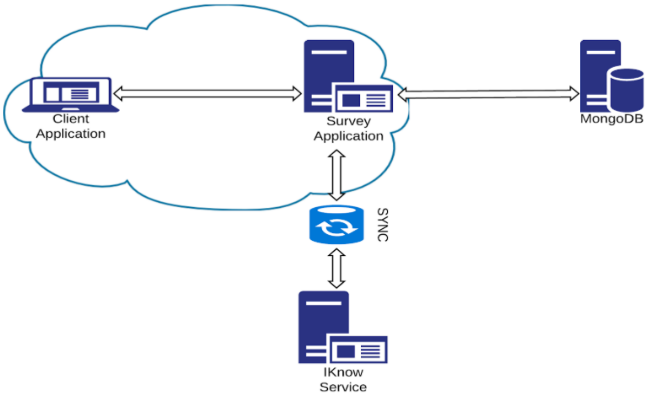
C. *Student surveys*

The faculty framework used to have this kind of system, but due to legacy frameworks and migrating to IDP authorization and authentication [13], the application was rebuilt using modern state-of-art frameworks such as Angular 6 and ASP.NET Core.[14] [15]

1) Workflow description: After completion of whole semester, the Faculty of Computer Science and Engineering opens schedule on the system for surveys, which allows all students to evaluate the professors who is teaching to them.The software requires from students login with their IKnow account,so if the login pass successfully, the user receives his own identifier (token) by the IKnow system. When each student logs in, the survey system performed synchronization with IKnow system which results by taking the appropriate courses. The appropriate courses are courses that the student listened in past semester for which the faculty has created a schedule. Once the whole process is carried out, the student has the opportunity to evaluate the professors for the relevant subjects. When a student chooses a particular subject for which he wants to evaluate the professors, the student gets a form he student gets a form to select the appropriate teaching assistants (if the subject is taught by more teaching assistants), while the professors are taken directly from IKnow system. Once the student has finished the process, he has the opportunity to answer a couple of questions about the teaching process that he follow ie to evaluate his professors and teaching assistants. After this step, the student has the opportunity to save his answers. When the student saves the answers,the database on which is connected this system, created appropriate record. The record contains data about the list of grades that the student filled in form for the appropriate course. In the end, the student is redirected to the home page where the courses of the semester for which he has created a schedule is displayed, and the corresponding subject for which the student answered is removed from that list.

There are two main building blocks which are shown on Fig. 1:

Fig. 1. Application architecture



2) System architecture: The implemented system architecture follows the interoperability standards considering the heterogeneity of the external services which are used for data harvesting. IKnow synchronization is made by using RESTful services. Login services are implemented by using Shibboleth protocol. The core architecture of the system is presented in Fig. represents a very simple and flexible coupled solution which is effortless to maintain and expand. The survey application is made by using ASP.NET Core technology which is intended for microservices architecture development. In the implementation of the front-end application, Angular 6 is used. Non-relational database MongoDb, is used as data storage. The reason that we decided to user MongoDB is its performances in write operations and large-scale abilities. The communication between client application and survey application is made by using RESTful services which rises the separability and functionality tier. We use the following synchronization processes:

- Students synchronization with iKnow per semester
- Courses synchronization with iKnow per student
- Teachers synchronization with iKnow per course

All database writes are anonymous. We do not keep track of the users which have filled the surveys in order to keep privacy. In the other side, we keep information about the user that he has completed the answering of the survey. This architecture provides the concept of federalization of survey services for all faculties in the University. The management and support are centralized, placed in FCSE, providing efficient control of software upgrades and improvements.

IV. Conclusion

Faculty of Computer Science and Engineering continues the development of e-platform for student and staff services by providing new e-services and their adaptation with machine interfaces to the central data repository. Such services provide simplification and acceleration of the Faculty administrative workflows by providing easy-to-use interfaces avoiding congestion and bottle-neck scenarios.In this paper we present novel services implemented in FCSE and UKIM as well. After the success implementation of the system for management of the process for diploma thesis defense, FCSE decides to implement an online solution for workflow automatization of the process for master thesis defense. The implementation of such process is more complex, but facilitates the user interactions. The main goal in implementation is adaptation of the same software and offering as a functional component to the other faculties in UKIM. Also, we develop a software for student survey answering considering the new legislations in teaching staff assessment. The main idea behind implementation of this software is to improve the concept of survey answering among the students and gathering the general opinion about the quality of the curricula and education globally at the faculty. All of the novel services provide scalable architecture. Also, inter-service communication is improved by adding microservice components in implementation of the new one. Central authorization, user management, the concept of single point of responsibility and interoperability improve the quality of e-services and facilitate the future upgrades with novel services.

References

[1] Beatriz Plaza. Google analytics for measuring website performance.

[2] Fabio Casati Daniela Grigori Hamid R. Motahari Nezhad Benatallah, Boualem and Farouk Toumani. Developing adapters for web services integration.

[3] M.P. Papazoglou. Service-oriented computing: Concepts, characteristics and directions.

[4] UDDI Consortium. Uddi executive white paper, nov. 2001.

[5] Microsoft sql server. https://www.microsoft.com/en-us/server-cloud/products/sql-server/.

[6] Microsoft web api. http://www.asp.net/web-api.

[7] E. Al-Masri and Q.H. Mahmoud. Investigating web services on the world wide web.

[8] H. Kuno V. Machiraju G. Alonso, F. Casati. Web services: Concepts, architectures, and applications.

[9] F. Casati B. Benatallah and F. Toumani. Web services conversation modeling: A cornerstone for ebusiness automation. ieee internet computing, 8(1), 2004.

[10] E. Pimentel J. Troya A. Vallecillo C. Canal, L. Fuentes. L. bordeaux et al. when are two web services compatible?. vldb tes'04. toronto, canada. 2004.

[11] Omar Benjellourn Ioana Manolescu Tova Milo Abitrboul, Serge and Roger Weber. ”active xml: Peer-to-peer data and web services integration.” inproceedings of the 28th international conference on very large data bases, pp. 1087-1090.

[12] E. Pimentel J. Troya A. Vallecillo C. Canal, L. Fuentes. L. bordeaux et al. when are two web services compatible?. vldb tes'04. toronto, canada. 2004.

[13] I. Dimitrovski V. Dimitrova K. Mishev, A. Stojmenski and I. Chorbev. Cloud services for faculty workflow automatization.

[14] J Lowy. Programming wcf services.

[15] B. Green and S Seshadri. Angularjs.

Investor platform for aggregating companies news articles aided by text sentiment analysis

Filip Spasovski, Kostadin Mishev, Ana Gjorgjevikj, Dimitar Trajanov
”Ss. Cyril and Methodius” University in Skopje
Faculty of Computer Science and Engineering
”Rugjer Boshkovikj” 16, 1000 Skopje, Republic of Macedonia

Abstract—**Nowadays, news aggregators become efficient tools for data aggregation which facilitate the comparison of different points of view established per publisher . Considering the fact that each day, Internet produces 2.5 quintillion bytes of data from different sources , we need different representation of textual data, aided by visual elements, in order to exclude the most important information presented in the article.**
In this paper, we present a world news and tweets platform which aggregates news articles related to big companies from different data sources including Twitter related to companies world-wide. Also, we calculate the sentiment of articles, RSS feeds and tweets and compare it with stock price data. Additionally, we give different view of the current news related to companies by extracting the most important keywords presented in the articles by using NLP tool-kits.
This platform is intended to help investors in their decision-making process for future investments giving a one-click solution for near-past trend of company development.

Keywords: Investor tool, Sentiment analysis, News data aggregation, News data analysis, Cloud news service.

I. Introduction

Data published on Internet increases each day [1]. The quantity of presented textual information, in a form of news article, can not be even followed by the readers due to the enormous number of news agencies and portals which exists on freely available Internet. This exacerbates the information flow to get directly injected into information consumers.

Sentiment analysis or opinion mining presents an automated process of opinion classification for a given text or speech. Due to the large number of articles published each day, we need automated tools which are able to perform sentiment analysis as a key tool for making sense of that data [2]. This has allowed companies to get key insights and automate all kind of processes like monitoring the consumer opinion for a product by sentiment analysis of comments left on social networks. The tone received as a result of sentiment analysis, can be used as an input in further company decisions and risk assessment. This facilitates the market research and analysis and improves the productivity of the company. With the growing demand for sentiment analysis tools in financial and economic applications, it is increasingly important to pay attention to the ability of the models to capture the domain-specific use of language.

II. Investor platform

In this paper, we present an investor platform which can be access on the following link: http://companies.b1.finki.ukim.mk/. We provide services for data aggregation from different sources gathering the data in real-time. Also, we calculate the sentiment of current news articles and tweets by using Stanford NLTK [3] and Loughran-Mcdonald financial dictionary [4].

A. Data sources

During the process of developing a solution for near-past trend of company development visualization, we strove to aggregate information from multiple sources.

1) Dbpedia: Dbpedia is a crowd-sourced community effort to extract structured content from the information created in various Wiki-media projects [5]. This structured information resembles an open knowledge graph (OKG) which is available for everyone on the Web. A knowledge graph is a special kind of database which stores knowledge in a machine-readable form and provides a means for information to be collected, organized, shared, searched and utilized.
Using the capabilities which Dbpedia provides, we provide to user the ability to explore data about thousands of companies worldwide.

2) GDELT: GDELT is the largest, most comprehensive, and highest resolution open database of human society ever created [6]. Creating a platform that monitors the world’s news media from nearly every corner of every country in print, broadcast, and web formats, in over 100 languages, every moment of every day and that stretches back to January 1, 1979 through present day, with daily updates, required an unprecedented array of technical and methodological innovations, partnerships, and whole new mindsets to bring this all together and make it a reality.

3) World Trading Data: World Trading Data is platform which offers a powerful and easy way to access market data from all over the world. It provides real time, or historical data for any stock, index, mutual fund or forex.

4) Twitter API: Twitter data is the most comprehensive source of live, public conversation worldwide. Twitter REST APIs enable programmatic analysis of data in real-time or back to the first Tweet in 2006. They enable the user to have an insight into audiences, market movements, emerging trends, key topics, breaking news, and much more.

Among the other information about the company that we provide to the user, we also present them an information about the positive, neutral and negative sentiment of the tweets about that specific company. This is accomplished through the application of the sentiment analysis process over the data collected from the Twitter API.

5) SEC litigations API: SEC litigations [1] web page provides an access to information about litigations in which our companies of interest took part.

B. Architecture

The architecture of our application represents a very simple, flexible and loosely coupled solution which is effortless to maintain and expand. There are three main building blocks which are shown in Fig. 1.

1) API gateway: The api gateway is one of the fundamental parts of the microservice architecture. This part encapsulates multiple operations which include: authentication, request routing and load balancing, cache management and others.

2) Microservices: Microservices represent the core building blocks of a microservice architecture. They are non-agnostic, often with a small functional scope, encompassing logic with specific processing and implementation requirements.
Our solution consists of six micro-services:

- **Company search** - holds the logic which enables to user to search through thousands of companies worldwide
- **News** - responsible for processing the data from the GDELT project, retrieving the valuable events about the company and crawling the news articles from the WEB
- **Word cloud** - applies NLP methods to generate a word cloud based on the news articles about the company
- **Stock market** - responsible for retrieving the stock information about the company
- **Twitter sentiment** - performs a sentiment analysis over the tweets about the company
- **Litigations** - encapsulates the operations related to the litigations in which the company was involved

3) Data sources: The data source layer consists of all the data sources mentioned in the previous section, excluding the WEB. This source is mentioned as the result of the crawling operations that the News micro-service performs.

III. Platform overview

In this section we are going to present the main functionalities of our platform, through the client application and her two main components.

A. Company search component

The company search component represents the entry point of the application. This very simple component, shown in Fig. 2. It consists single text input field and enables the users to search and find the companies of their interest.

After the user enters the company name in the input field, a request is sent to the company search micro-service. This service queries the Dbpedia's information about companies that match the given criteria, through the Dbpedia's Lookup API. When the results are ready, they are served to the client in a simple list shown in Fig. 3.
The list of company results enables the user to navigate to the company details component, with a single click on either of the list items.

B. Company details component

After the user clicks on a search result, the application navigates to the company details component. This component is the main building block of the application and contains all the information that we want to provide to the user. To simplify the process of describing all the data that is shown here, we will break down this component in smaller inner components which represent a valuable information.

1) News component: This component represents a list of recent news related to the company, which is delivered by the news micro-service. After the service accepts the request from the client, it filters the valuable information from the recent GDELT events and starts with the process of crawling the news articles from their respective Web sites. When the process is finished, the news articles are delivered to the client. The resulting component is shown in Fig. 4.

2) Word cloud component: shows a generated word cloud from the combined content of the resulting news about the company. Word clouds are a novelty visual representation of text data, typically used to depict keyword meta-data on websites, or to visualize free form text. This component which is shown in Fig. 5 enables the users to have a quick summary about the recent news articles, without reading them separately.

3) Stock market component: - provides the user with a detailed graphical information about the stock pricing of the respective company. The user is able to specify the month and the year, as well as the type of the stock price (open, close, high, low). The x-axis represents the day of the month, while the y-axis represents the stock price value, typically in USD. The pairs of the stock price values in each day of the month form the green line shown in Fig. 6. This line provides the user with a visual representation of the stock price values through the specified year and month. The vertical red lines represent the litigations in which the specified company took part in and lost. Combining the both pieces of information, we wanted to help the user to visualize the correlation between them.

4) Twitter sentiment component: This component shows the result of the twitter sentiment analysis, performed over the tweets related to the company. In the Fig. 7 we can effortlessly visualize what is the people's current opinion about the specific company.

5) Litigations component: This component crawls specific litigations whereas the companies take part. The Security and Exchange Commission (SEC) in USA publishes them on its web site. When the company loose a litigation, it is obvious that their reputation decreases. This may be an indicator of stock price change and may be important information which

[1] https://www.sec.gov/litigation/litreleases.shtml

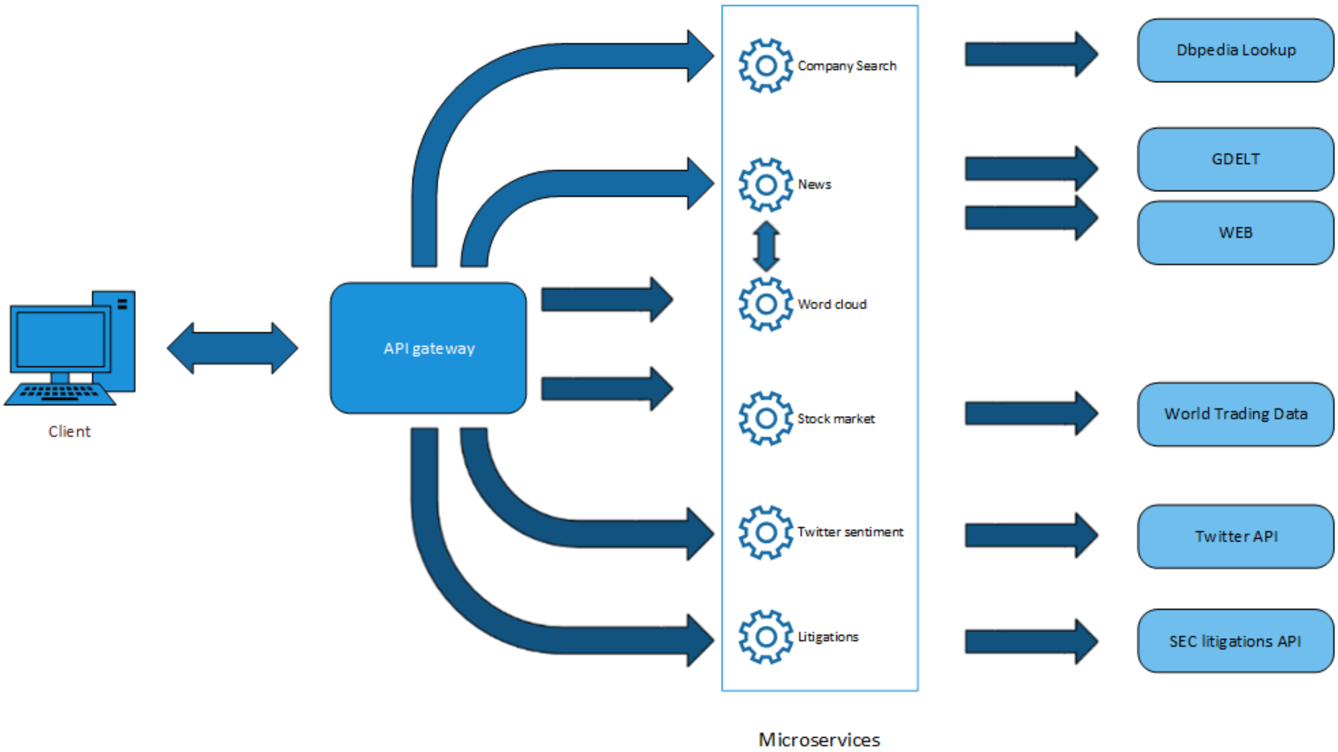


Fig. 1. Application architecture

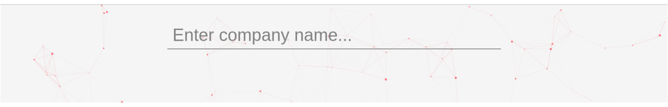
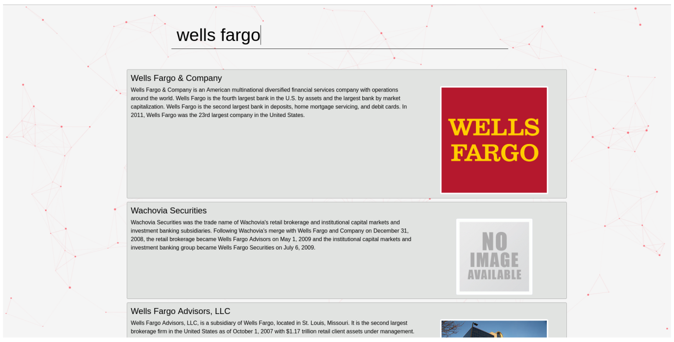


Fig. 2. Company search component



wells fargo

Wells Fargo & Company

Wells Fargo & Company is an American multinational diversified financial services company with operations around the world. Wells Fargo is the fourth largest bank in the U.S. by assets and the largest bank by market capitalization. Wells Fargo is the second largest bank in deposits, home mortgage servicing, and debit cards. In 2011, Wells Fargo was the 23rd largest company in the United States.

Wachovia Securities

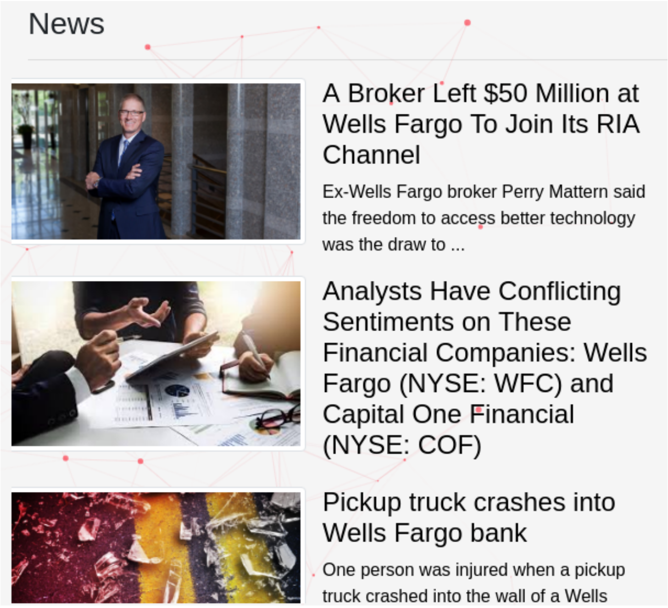
Wachovia Securities was the trade name of Wachovia's retail brokerage and institutional capital markets and investment banking subsidiaries. Following Wachovia's merge with Wells Fargo and Company on December 31, 2008, the retail brokerage became Wells Fargo Advisors on May 1, 2009 and the institutional capital markets and investment banking group became Wells Fargo Securities on July 6, 2009.

Wells Fargo Advisors, LLC

Wells Fargo Advisors, LLC, is a subsidiary of Wells Fargo, located in St. Louis, Missouri. It is the second largest brokerage firm in the United States as of October 1, 2007 with $1.17 trillion retail client assets under management.

Fig. 3. Company search result

should be presented to the investor.
We crawl all litigation from the SEC web page by using the litigation feed. In order to obtains successful crawling of HTML pages, we use python library NewsPlease which provide efficient extraction of text from HTML page.
Next, we use Python library Spacy [2] [7] to extract the companies presented in the text. Afterwards, we make time-stamps alignment with stock price data history for the same



News

A Broker Left $50 Million at Wells Fargo To Join Its RIA Channel

Ex-Wells Fargo broker Perry Mattern said the freedom to access better technology was the draw to ...

Analysts Have Conflicting Sentiments on These Financial Companies: Wells Fargo (NYSE: WFC) and Capital One Financial (NYSE: COF)

Pickup truck crashes into Wells Fargo bank

One person was injured when a pickup truck crashed into the wall of a Wells

Fig. 4. News component

company. The results of the alignment are presented on Fig. ??. The red vertical lines present the date when a litigation process has been finished.

IV. Conclusion

Our solution targets the potential company investors, providing them with guidance about their next decision-making

[2] https://www.spacy.io/

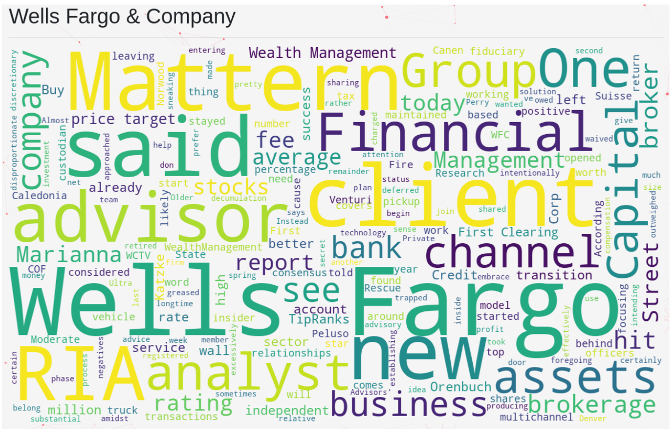
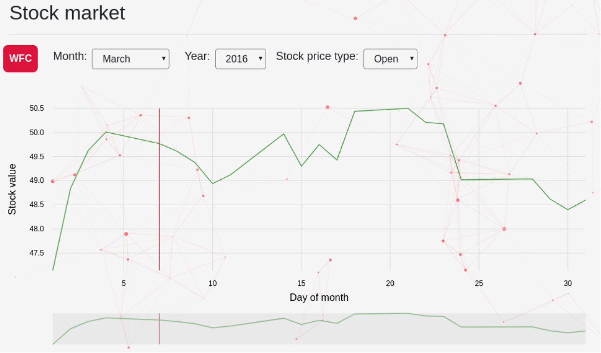


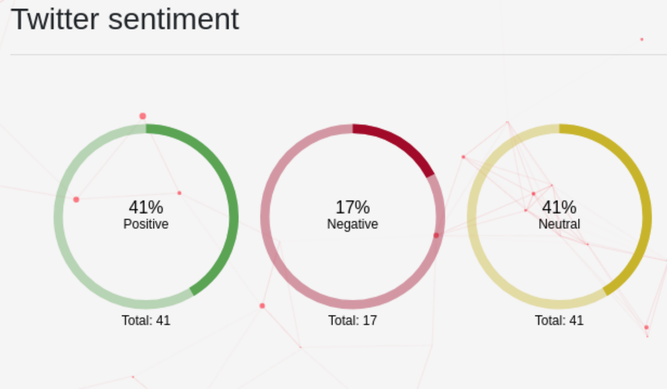
Fig. 5. Word cloud component



Stock market

WFC Month: March Year: 2016 Stock price type: Open

Fig. 6. Stock market component



Twitter sentiment

Fig. 7. Twitter sentiment component

process. Presenting the investors with an information from different sources, we aimed at redesigning the procedure through which they pass whenever they want to place an investment. By overcoming the obstacles, which occurred during the process of collecting, connecting and transformation of data, we managed to build a stable platform that is easy and ready to use.

References

[1] Payam Barnaghi, Wei Wang, Cory Henson, and Kerry Taylor. Semantics for the internet of things: early progress and back to the future. *International Journal on Semantic Web and Information Systems (IJSWIS)*, 8(1):1–21, 2012.

[2] Susan Athey and Markus Mobius. The impact of news aggregators on internet news consumption: The case of localization. In *Workshop on the Economics of Web Search and Social Network. Sixth ACM International Conference on Web Search and Data Mining 2013*, page 2, 2012.

[3] Christopher Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven Bethard, and David McClosky. The stanford corenlp natural language processing toolkit. In *Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations*, pages 55–60, 2014.

[4] Tim Loughran and Bill McDonald. When is a liability not a liability? textual analysis, dictionaries, and 10-ks. *The Journal of Finance*, 66(1):35–65, 2011.

[5] Sören Auer, Christian Bizer, Georgi Kobilarov, Jens Lehmann, Richard Cyganiak, and Zachary Ives. Dbpedia: A nucleus for a web of open data. In *The semantic web*, pages 722–735. Springer, 2007.

[6] Kalev Leetaru and Philip A Schrodt. Gdelt: Global data on events, location, and tone, 1979–2012. In *ISA annual convention*, volume 2, pages 1–49. Citeseer, 2013.

[7] Jinho D Choi, Joel Tetreault, and Amanda Stent. It depends: Dependency parser comparison using a web-based evaluation tool. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, volume 1, pages 387–396, 2015.

Movie Review Sentiment Analysis

Jana Kuzmanova
Faculty of Computer Science and Engineering
Ss. Cyril and Methodius University
Skopje, North Macedonia
jana.kuzmanova@students.finki.ukim.mk

Ana Madevska Bogdanova
Faculty of Computer Science and Engineering
Ss. Cyril and Methodius University
Skopje, North Macedonia
ana.madevska.bogdanova@finki.ukim.mk

***Abstract*—An algorithm for the sentiment analysis of movie reviews was implemented and evaluated using various tools, methods and corpora from nltk with the purpose of determining whether the results differ based on the length of the text being evaluated, as well as the types of features used for classification. The algorithm can classify reviews on the document and sentence level. The results show that these factors do affect the performance of the classifier.**

I. Introduction

Sentiment analysis is a field of text mining which focuses on extracting opinions about products or other opinion targets from textual data. The goal is usually to aggregate opinions about the product or topic being evaluated, but also to identify fake reviews[1], [2]. Formally, an opinion can be defined as a quadruple (g, s, h, t) of the sentiment target g, sentiment s, opinion holder h and the time t when the opinion was expressed[1]. Depending on the granularity of the analysis being performed, the opinion target can be replaced by an entity e that represents the whole target and aspect a which represents the specific aspect of the entity that's being evaluated in the given snippet of the text. The formally defined goal of sentiment analysis is to find all of these quadruples or quintuples in the text.

Sentiment analysis can be performed on various levels of granularity. The document level treats the whole source text from which the opinion is being extracted as a whole. On this level, we assume that the given review or comment evaluates one entity and we only extract one opinion, positive or negative, about it. This assumption often isn't applicable. Even reviews of a single product can evaluate different aspects of this product, compare the opinions of multiple opinion holders on it, or compare the product with other products. Sentence-level sentiment analysis aims to address some of these issues by analyzing the sentiment of each sentence, and if wanted, aggregating it to determine the polarity of the whole document that the sentences are part of. On this level, we perform subjectivity classification to determine whether the sentence is subjective before trying to determine its polarity. However, a single sentence can still express multiple opinions. The finest level of granularity in sentiment analysis is the aspect level. Here, both the entities and their aspects, as well as the opinions expressed about them are extracted. This is the most realistic type of sentiment analysis, however, it is very challenging as it consists of many subproblems, including entity extraction and classification.

Sentiment analysis is a classification problem. In the broadest sense, it uses standard classification algorithms such as Naive Bayes or SVM with appropriate features in order to determine the most likely class label for the given instance. The problem usually is to try to classify the instance into one of two classes, positive or negative for the document level, whereas for the sentence level it can be defined as a three-class problem that also includes a neutral class label, or two two-class problems by first classifying the sentence as subjective or objective. The choice of features is crucial to the performance of the classification algorithm being used. The simplest type of features are unigrams, which actually perform fairly well. Other types of features that can be used are bigrams, part-of-speech tags, word frequency, as well as sentiment words which express opinions and sentiment shifters, such as negations[1].

In this paper, we perform document- and sentence-level sentiment analysis on movie reviews and we compare the performance of the Naive Bayes algorithm on various document lengths and using various features, including unigrams, bigram collocations, and unigrams without stopwords.

II. Methods and materials

The implementation presented in this paper uses various tools from the nltk library. The Naive Bayes classifier from the nltk.classify package was used for classification. Other classifiers included in this package are the decision tree and the maximum entropy classifier. It also contains the scikitlearn module, which wraps scikit-learn and allows for any of the classifiers defined in it to be used. The naivebayes module implements the Naive Bayes classification algorithm, which assumes independence of all features. The module contains the classmethod *train*, as well as *classify*, which classifies a given instance, and *show_most_informative_features*, which shows which features are the most distinctive, and which class is indicated by their presence[3].

The training corpora used are the *subjectivity*, *movie reviews* and *sentence polarity* from nltk.corpus. The subjectivity corpus contains 10000 labeled sentences with an equal distribution in the two classes, objective and subjective. The samples in this dataset were labeled automatically, under the assumption that all sentences from the Rotten Tomatoes pages are subjective, and all sentences in the IMDb plot summaries objective[4], [5]. The movie reviews dataset is made up of reviews class-labeled on the document level. It contains a total of 2000 reviews, again equally split into positive and negative. These files are also automatically labeled, by detecting an explicit rating in the review, and extrapolating its polarity from it[5], [6]. This corpus can also be viewed on the sentence level, in which case it contains a slightly unbalanced class distribution of 36037 positive and 35495 negative sentences. The sentence polarity corpus contains 5331 positive and 5331 negative sentences. The data is taken from the Rotten Tomatoes, labeling sentences and snippets from reviews marked as 'fresh' as positive, and those from reviews marked as 'rotten' as negative[14], [15].

III. Implementation

Since the movie reviews corpus only has a positive and a negative class, the algorithm is implemented as a two-step sentiment analysis system which first classifies the sentences of the input document as subjective or objective, and then classifies the subjective sentences as positive or negative. The implementation uses the Naive Bayes classifier from the nltk.classify module, together with tools and utilities from nltk.sentiment. Pre-processing is performed using the nltk.tokenize module.

The only pre-processing step taken is using the sentence tokenizer to split the input into sentences, followed by using the WordPunctTokenizer to represent these sentences as a list of words and groups of punctuation signs. The result is a list of lists. Each of the inner lists is passed to an instance of SentimentAnalyzer which has been trained with a classifier and a training set. The training set is comprised of tuples whose first member is a list of words and the second a class label, and it contains an equal number of instances of both classes, while the classifier uses unigram and bigram features extracted with either built-in or user-defined functions.

Training data for the subjectivity classification is comprised of n sentences of each of the two classes of the subjectivity corpus. Next, an instance of SentimentAnalyzer which contains many of the function used in this implementation is initialized. Then the *mark_negation* function is applied to the sentences. This method appends the suffix _NEG to all words in the sentence which appear between a negation and a punctuation mark. Then use the function *all_words* is used to return a list of all words in all sentences in the training set. This list or the list of sentences with negation handling is then passed to the feature extraction functions.

In this case the built-in unigram and bigram collocation feature extractors are used, as well as a user-defined unigram feature extractor which filters out stopwords. The *unigram_word_feats* function takes a list of words and a minimum frequency as arguments. It calculates a frequency distribution for the words passed in the first argument, and returns those whose frequency is higher than the value of the second argument. The user-defined unigram extraction function *all_words_nosw* takes the list of sentences with marked negation as input and filters out the words that are present in nltk.corpus.stopwords. It then continues as the built-in *all_words* function, returning the words with a frequency higher than the given minimum frequency. The goal is to create a smaller and more efficient feature set with fewer words that are likely to appear often in both classes and therefore aren't very informative for the classifier. Both of these feature sets are added to the sentiment analyzer using the unigram feature extractor with a different list as its unigrams argument. Another type of features taken into consideration are the bigram features. The most common and meaningful bigrams are extracted and then the bigram extractor is added to the sentiment analyzer with the list of bigram collocations as argument. A function which extracts part-of-speech tags to be used as features and the extractor which adds these features to the classifier are also implemented. The *pos_tags* function returns all part-of-speech tags found in the training set using nltk's POS tagger. Due to the way it's implemented, this automatically also includes punctuation as features. The extractor then returns a dictionary with information on whether each of these tags is present in the current document.

After all feature extractors have been added, they are applied to the sentiment analyzer and the current training set. This step applies all feature extractors we've added to the analyzer to the sentences in the training set, and checking whether each of the tokens in the feature sets is present in the given sentence. The resulting list of tuples indicating the presence or absence of words and the label for the given instance forms the final training set. Finally, the sentiment analyzer can be trained by passing the training function of the used classifier, here NaiveBayesClassifier and the training set. The returned classifier can then be used to classify sentences as objective or subjective.

The second classifier which determines the polarity of a review is built in a similar way. Since the movie_reviews corpus classifies whole documents as positive or negative, as well as classifying the sentences belonging to these reviews as positive or negative based on the polarity of the review, a training set can be created the same way as for the subjectivity corpus, or all of the words belonging to each review can be used to train the classifier on the document level. The process then continues the same way as for the previous classification. When using the sentence_polarity corpus, the classification is always performed on the sentence level.

Finally, both of the classifiers can be used to classify the input reviews. The classifier's classify function is called with the tokenized sentence as its argument. The result is returned as a dictionary with two keys, 'pos' and 'neg' with its values being lists of the sentences classified as belonging to the respective class. The document can also be classified on the document level by tokenizing the whole input and directly classifying it as positive or negative, foregoing the subjectivity classification.

IV. Evaluation

The performance of the presented implementation is evaluated with varying feature sets and functions, length of reviews being classified, document vs. sentence-level training and classification, and number of instances of the corpus being used

for training. In each case, the full 10000 sentence subjectivity corpus and a Naive Bayes classifier are used.
The system was tested on the following types of reviews:

- long positive reviews
- long negative reviews
- short positive reviews
- short negative reviews

Both professional and reviews written by internet users are used in these tests. The latter can contain improper punctuation, typos, and other speech or writing patterns which aren't likely to be used by professional film critics, however for now this is ignored.
The features used for training are also considered. The classifier was trained with the following features:

- unigrams using all words
- unigrams without stopwords
- bigram collocations
- part-of-speech tags
- unigrams and part-of-speech tags

The following tests only compare the performance of the sentiment classifier based on the granularity level on which training and classification is performed. The subjectivity classifier always uses sentence-level training and classification. All tests were run on a classifier trained on 500 instances of each class for the document-level training and 15000 sentences for the sentence-level training, or half of the full data set, and applied to reviews from all classes used in the previous tests. The features used were all unigrams with a frequency greater than 4.

Type of classification	Accuracy
Document-level training, document-level testing	4/5
Document-level training, sentence-level testing	2/5
Sentence-level training, sentence-level testing	3/5

TABLE I: Number of correctly classified reviews by testing and training level

The classifier performs best on the task that it was made for - classifying whole movie reviews on the document level. It often classifies positive sentences as negative. This might be because the sentences in this corpus are tagged based on the whole review, so if a negative review contains a positive sentence, it will be labeled as negative. Since reviews can often be mixed or point out both the good and the bad aspects of the movie despite having a clear opinion can lead to skewed results.
The following tests were run on a classifier trained on 5331 instances of sentences in the movie reviews dataset. The classifier used unigram features and the classification was performed on the sentence-level. The table shows the number of sentences in the review classified as positive and negative, and the remaining sentences are considered objective and therefore don't get classified further.

Type of review	Sentences classified as positive	Sentences classified as negative
Long, positive	19/75	26/75
Long, negative	6/31	10/31
Short, positive 1	4/15	7/15
Short, positive 2	0/10	10/10
Short, negative 1	1/6	4/6
Short, negative 2	0/8	8/8

TABLE II: Number of sentences classified as positive and negative by type and length of review

The same tests were repeated for the sentence polarity dataset, which explicitly contains sentences tagged as positive or negative. This dataset consists of 5331 positive and 5331 negative sentences and the same number of training sentences was used in the previous test for easier comparison.

Type of review	Sentences classified as positive	Sentences classified as negative
Long, positive	20/75	25/75
Long, negative	3/31	13/31
Short, positive 1	9/15	2/15
Short, positive 2	2/10	8/10
Short, negative 1	0/6	5/6
Short, negative 2	1/8	7/8

TABLE III: Number of sentences classified as positive and negative by type and length of review

The classifier trained on the sentence polarity corpus gives better results. This isn't surprising, as it is more accurately labeled on the sentence level. However, this classifier also categorizes two of the positive reviews as negative. A closer look at the results shows that the sentences classified as negative in this case are mostly objective sentences that were wrongly classified as subjective. Among the sentences classified as negative are snippets such as *"i guess it comes down to a simple choice, really," he tells red.*, *he doesn' t get real worked up.* or *frank darabont wrote and directed the film, basing it on a story by stephen king.* that either summarize or quote the movie without evaluating it. Others, such as *interesting that although the hero of the film is the convicted former banker andy dufresne(tim robbins), the action is never seen from his point of view.* or *maybe it plays more like a spiritual experience than a movie.* also don't express an opinion despite being classified as subjective and negative. However, there are also sentences that are expected to be incorrectly classified, like *i think such movies are slower to sit through than a film like "shawshank," which absorbs us and takes away the awareness that we are watching a film.* or *roger deakins' cinematography is tactful, not showy.*, which positively compare the movie being reviewed with others, with the negative evaluation affecting the classification. This is a weakness of the bag-of-words approach, which can be resolved by adding features concerning sentence structure or part-of-speech tags. The second short positive review is also wrongly classified as negative by both classifiers. This review uses a lot of nonstandard grammar and punctuation, which likely contributes to the incorrect classification.
The evaluation of the performance of classifiers trained using various feature sets was performed on 5 reviews, 3 positive and 2 negative. All reviews from the movie reviews corpus for

the unigram features and 200 reviews for the unigrams with no stopwords, part-of-speech tags, and bigram collocations. A combination of unigrams and part-of-speech tags is also used, again with a training set of 200 reviews. The training and classification was performed on the document level.

Features used	Accuracy
unigrams (all)	4/5
unigrams (no stopwords)	3/5
bigram collocations	4/5
POS tags	1/5
unigrams + POS tags	4/5

TABLE IV: Number of correctly classified reviews by type of training features

The results show that using all unigram features is much more effective than the other two options. Even though the assumption was that filtering out the stopwords would make the classifier more effective, it's possible that the computational effort use to filter them out isn't worth it. However, using the bigram collocations as features was just as effective at classification as using the unigram features. This is offset by the fact that training the classifier using bigram features took much longer than using unigrams. Attempting to use part-of-speech tags on the document level proves to be ineffective - as the following table shows, the features don't have large discriminative power and quickly taper off to insignificant differences.

Feature	pos : neg ratio
contains(VBP) = False	5.0 : 1.0
contains(PRP$) = False	1.0 : 2.3
contains(VBD) = False	2.3 : 1.0
contains(,) = False	1.0 : 1.7
contains(UH) = True	1.0 : 1.3

TABLE V: Most informative POS features

On the other hand, the most informative features using the same training set and unigram features are the following:

Feature	pos : neg ratio
contains(avoids) = True	13.0 : 1.0
contains(astounding) = True	12.3 : .0
contains(slip) = True	11.7 : 1.0
contains(outstanding) = True	11.5 : 1.0
contains(ludicrous) = True	1.0 : 11.0

TABLE VI: Most informative unigram features

V. Conclusion

This paper takes a look at the task of classifying the polarity of movie reviews as positive or negative. Various tools, methods and corpora from nltk were used to implement an algorithm which can classify movie reviews on the document and sentence level. The features used were unigrams, unigrams without stopwords, bigram collocations and part-of-speech tags. An evaluation was performed to see if this algorithm performs differently when used on reviews of different length, as well as with different features. The results showed that longer reviews tend to have a larger number of incorrectly classified sentences, and therefore a larger chance of being incorrectly classified in general. This is mostly due to sentences that don't express an opinion being treated as subjective. The testing with various features showed that most of the features used performed well, but that part-of-speech tags on their own aren't powerful enough to correctly classify on the document level. In future work the algorithm will be expanded to use different classifiers, including ones that don't assume independence and the evaluation of the algorithm will be performed without examples that are likely to generate a false classification to see whether the performance will improve.

References

[1] B. Liu, *Sentiment Analysis and Opinion Mining*, G. Hirst, Ed. Morgan & Claypool, 2012.

[2] C. C. Aggarwal and C. Zhai, Eds., *Mining Text Data*. Springer, 2012.

[3] (15.03.2019) nltk.classify package. [Online]. Available: https://www.nltk.org/api/nltk.classify.html

[4] (20.03.2019) Subjectivity dataset. [Online]. Available: http://www.cs.cornell.edu/people/pabo/movie-review-data/subjdata.README.1.0.txt

[5] B. Pang and L. Lee, "A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts," in *Proceedings of the ACL*, 2004.

[6] (20.03.2019) Movie reviews dataset. [Online]. Available: http://www.cs.cornell.edu/people/pabo/movie-review-data/poldata.README.2.0.txt

[7] E. Cambria and A. Hussain, *Sentic Computing*. Springer, 2015.

[8] A. Kao and S. R. Poteet, *Natural Language Processing and Text Mining*. Springer, 2007.

[9] D. J. Olsher, "Full spectrum opinion mining: Integrating domain, syntactic and lexical knowledge," *IEEE 12th International Conference on Data Mining Workshops*, pp. 693–700, 2012.

[10] M. Allahyari, S. Pouriyeh, M. Assefi, S. Safaei, E. D. Trippe, J. B. Gutierrez, and K. Kochut, "A brief survey of text mining: Classification, clustering and extraction techniques," *arXiv preprint arXiv:1707.02919*, 2017.

[11] I. H. Witten, "Text mining," 2004.

[12] E. Alpaydın, *Introduction to Machine Learning*, 2nd ed., T. Dietterich, Ed. The MIT Press, 2010.

[13] (15.03.2019) nltk.sentiment package. [Online]. Available: https://www.nltk.org/api/nltk.sentiment.html

[14] (23.03.2019) Sentence polarity dataset. [Online]. Available: http://www.cs.cornell.edu/people/pabo/movie-review-data/rt-polaritydata.README.1.0.txt

[15] B. Pang and L. Lee, "Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales," in *Proceedings of the ACL*, 2005.

Named Entity Discovery for the Drug Domain

Nasi Jofche
Faculty of Comp. Sci. and Eng.
Ss. Cyril and Methodius University
Skopje, North Macedonia
nasi.jofche@finki.ukim.mk

Milos Jovanovik
Faculty of Comp. Sci. and Eng.
Ss. Cyril and Methodius University
Skopje, North Macedonia
milos.jovanovik@finki.ukim.mk

Dimitar Trajanov
Faculty of Comp. Sci. and Eng.
Ss. Cyril and Methodius University
Skopje, North Macedonia
dimitar.trajanov@finki.ukim.mk

***Abstract*—Medical datasets that contain data relating to drugs and chemical substances, in general tend to contain multiple variations of a generic name which denotes the same drug or a drug product. This ambiguity lies in the fact that a single drug, referenced by a unique code, has an active substance which can be known under different chemical names in different countries, thus forming an obstacle during the process for extracting relevant and useful information. To overcome the issues presented by this ambiguity, we developed a scalable, term frequency based data cleaning algorithm, that solely uses the data available in the dataset to infer the correct generic name for each drug based on text similarities, thus forming the roots for building a model that would be able to predict generic names for related and previously unseen drug records with high accuracy. This paper describes the application of the algorithm towards the cleaning and standardization process of an already populated drug products availability dataset, by representing all of the variations of a substance under a single generic name, thus eliminating ambiguity. Our proposed algorithm is also evaluated against a Linked Data approach for detecting related drug products in the dataset.**

***Index Terms*—Named Entity, Data Cleaning, Text Similarity, Drug Data, Drugs.**

I. Introduction

The drug product availability dataset, known as LinkedDrugs and available through the Global Open Drug Dataset (GODD) application [3, 21], contains crawled data on drug products registered in a large set of countries around the world, along with information about their respective active substances [4] and unique ATC codes [2]. The dataset, due to its nature, is prone to ambiguity related to multiple variations for a single active substance for a drug product. This ambiguity refers to the fact that a single drug might have an active substance – also referred to as *generic name* – which is known under a different name in different countries – e.g. the chemical substance Paracetamol is also known as Paracetamolum, Acetaminophen, N-acetyl-para-aminophenol, etc. [5], all categorized under the same unique ATC code. This ambiguity in the dataset might lead to disorienting results while trying to extract useful information from the data, thus can present an obstacle that needs to be eliminated during the data preprocessing phase [14]. This is important because different drug products can be labeled as *related* if they have the same active substance (generic name). Then, these related drug products can be used as alternatives of each other in various real-world cases. Therefore, the ability to correctly identify related drug products even when their active substances are not present in the dataset, or have non-matching values, is of high importance.

Besides the application of the widely known steps related to the data preprocessing phase for a dataset, such as handling missing values [11] which is a common scenario in our case, specific drug domain-based preprocessing and cleaning rules are required as well.

This paper presents a preprocessing algorithm that solely uses the available data from the dataset to infer the correct active substance (generic name) for a drug, thus reducing ambiguity. Since the dataset has a significant amount of missing values for the active substances of the drug products, the algorithm is based on a combination of text similarity metrics [15, 17] and ATC code equality. The algorithm consolidates the active substances of all groups of related drug products, by setting them to or replacing them with the most common text variant of the active substance of the given group of drug products.

For the purpose of testing the algorithm, two text similarity techniques were used: cosine similarity and Levenshtein distance. The algorithm is scalable and supports usage of other text distance metrics, as well. Its purpose is to provide preliminary results that build the roots of a future model that can be used to further discover named entities from previously unseen drug related data. The thorough algorithm description, as well as the obtained results are given in the following sections. Finally, the accuracy of our algorithm is tested against an entity detection approach based on Linked Data that detects all similar drug names in order to extract the active substance (generic name).

II. Related Work

Multiple efforts have been made into assessing the drug availability across different countries under consolidated datasets. The analysis in [9] shows a specific subset of drugs and their availability throughout different countries in Europe. On the other hand, the analysis made by [26] indicates the need for increased availability of drugs in 11 countries of the Asia Pacific Region.

The efforts made by [19, 20] use the Linked Data approach to consolidate drug product data in Macedonia, and then on a global scale [21]. These approaches provide a global overview of the drug products which are registered and sold in different countries, and the provide the ability to identify and analyze related drug products across and between countries. We use some of the approaches described in these papers to compare the accuracy of our proposed algorithm, later in this paper.

These efforts, as well as other similar research works, such as [16], present solutions based on Linked Data that reduce the ambiguity related to drugs branded in multiple names, overcoming the named entity ambiguity obstacle.

Other previous research efforts are focused on named entity recognition and detection from plain text or different datasets. Such different approaches for named entity recognition are described in [13, 23, 24], leading to diverse entity detection approaches in regard to the business domain [10, 18, 22, 25]. The different techniques analyzed for named entity detection are tightly coupled to the domain that they are applied to.

On the other hand, recent advances in this field indicate high accuracy while applying entity detection and linking algorithms on language independent datasets, as discussed in [12]. This proposed algorithm can be trained on one language and is shown to perform well on other languages without any change. It achieves the entity linking based on different kinds of language independent features and a discriminative ranking function.

III. Generic Names Dataset

In order to assess the accuracy of our algorithm, we created a dataset of generic names by extracting drug related information from DBpedia [1]. This dataset contains information about generic names for drugs, i.e. active substances, accompanied by the respective ATC codes in the following format:

Generic Name	ATC Code	Similar Substances
Paracetamol	N02BE01	Acephen:Acetaminophen...

The information in the dataset was gathered by querying the DBpedia endpoint using a custom SPARQL [6] query, focusing on the wikiPageRedirects property [7] to extract all different names under which a substance is known. The query returned 1,675 records which we used to test the accuracy of our algorithm, i.e. its ability to detect the correct generic names.

IV. Algorithm Application & Results

The first step in the process of the dataset analysis was focused on drug product groupings by their respective countries, in order to extract the information for the total number of initial generic names, both correct and incorrect. The dataset of generic names was used to assess this accuracy, by comparing the active substance (generic name) value of each drug product from our LinkedDrugs dataset to the full list of generic names. Table I gives an overview on the initially correct generic names for the respective countries, by analyzing a total of 10 countries.

TABLE I
Initial Generic Name Accuracy of Drug Products by Country

Country	Incorrect	Correct	Correct [%]
DK	6,410	2,535	28.33
BIH	2,935	1,506	33.91
AZ	2,913	1,529	34.42
US	13,255	7,159	35.06
SL	1,468	933	38.85
BE	4,554	4,263	48.34
MT	3,193	3,506	52.33
CY	1,951	2,587	57.00
FIN	2,641	3,595	57.64
MK	983	2,378	70.75

Due to the large size of the version of the LinkedDrugs dataset being analyzed – 125.424 drug product records at the time of analysis – and the complexity of the algorithm being O(n^2), we decided to assess the algorithm accuracy based on smaller subsets chosen randomly. The algorithm was applied using both cosine similarity and Levenshtein distance, in combination with the available ATC codes for identifying similar substances. Each drug product of the dataset was compared respectively by measuring text distance and only counting distances above a specified threshold, thus forming a sparse vector with a length same as the selected subset. In addition to that, each drug product with the same ATC code was also analyzed to create an additional sparse vector, which was combined in a union relationship with the frequency vector. The resulting vector was used to replace each entity with the most frequent one, thus detecting the most common generic name for the substance which is the active ingredient for the given group of drug products. The resulting cleaned dataset records were compared to the dataset of generic names, to assess the generic name recognition accuracy. The obtained results are given in Fig. 1 and Fig. 2, using cosine and Levenshtein metrics respectively, which indicate an increase in accuracy as the subset size increases. For each subset size we chose five random subsets, and then averaged the results.

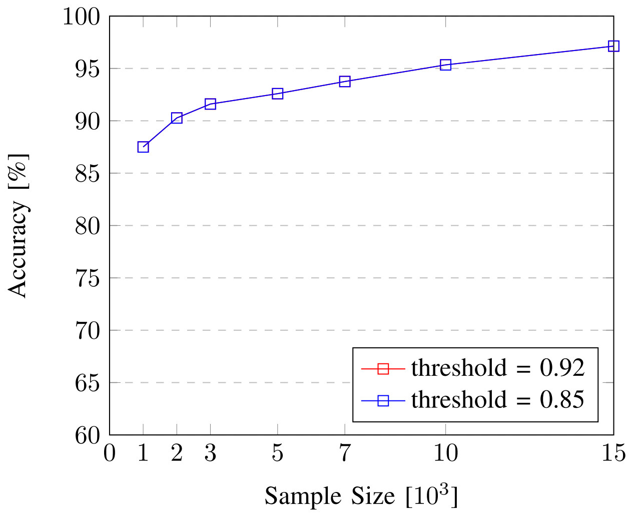


Fig 1. Accuracy Assessment using Cosine Similarity

The obtained results indicate high accuracy, which is attributed to the selected subsets containing mixed data from different countries. We can conclude that both similarity thresholds that were chosen while assessing the cosine similarity performance, show the same results.

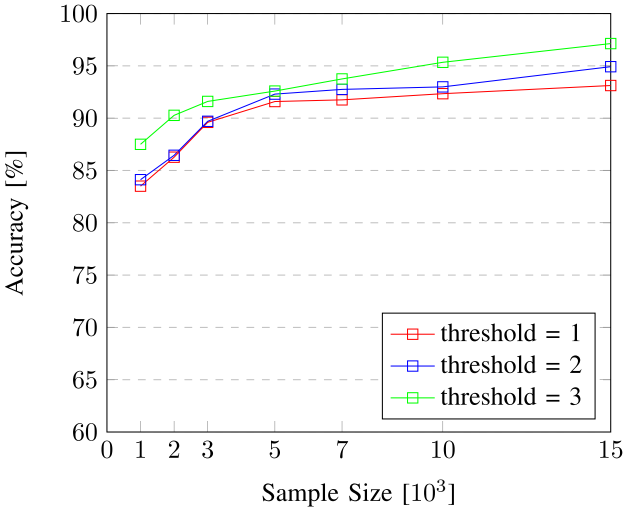


Fig 2. Accuracy Assessment using Levenshtein Distance

The algorithm's performance using the Levenshtein distance metrics with variable distance thresholds indicates high accuracy as well. It is worth noting that for smaller data subsets and a smaller threshold, the obtained accuracy was slightly lower – a result related to the missing ATC code values for drug products. While using the distance threshold = 3, the obtained accuracy was the same as the accuracy obtained while using the cosine similarity.

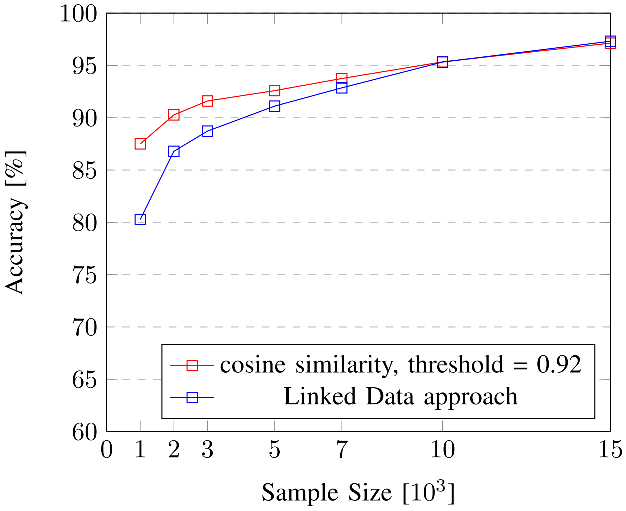


Fig 3. Comparison to the Linked Data Approach

V. Comparison to the Linked Data Approach

In this section we evaluate the accuracy of our proposed algorithm against the Linked Data approach for finding similar drug names in the dataset [21]. We use OpenRefine [8] to apply a transformation script to our dataset, in order to identify similar drugs and infer the generic name by reconciling against drug entities from DBpedia. The obtained results are compared to the results obtained from our algorithm by using the cosine similarity with similarity threshold = 0.92, and are given in the chart in Fig. 3. These results indicate that our algorithm outperforms the Linked Data approach for smaller dataset subsets, while it is slightly outperformed for larger subsets.

VI. Future Work

The described algorithm is part of an ongoing research, serving as the basis for a highly accurate model capable of predicting the generic names of drug products from data that it has not encountered before, with a sole purpose of extracting useful information from the dataset and using the obtained information for making country-related predictions regarding the drug availability.

After this highly accurate, domain-specific cleaning process, the next step is building and improving a neural network that will correctly classify the incoming data from the crawlers. Since the crawlers used in the GODD applications are continuously enriching the dataset, applying the algorithm to the entire dataset every time a new record gets extracted is inefficient, thus a neural network that will be trained using the cleaned and accurate data obtained from the algorithm would be a highly scalable solution.

The same algorithm can further be used to detect company name entities, as well, which are also available in the dataset under different names.

VII. Conclusion

Besides the challenges faced when applying the common cleaning steps to a dataset, applying domain-specific cleaning techniques is a challenge of its own. In this paper we presented a highly accurate algorithm focused on cleaning a drug product availability dataset by detecting named entities and standardizing generic names of substances across all records. The described algorithm uses text similarity metrics: cosine similarity and Levenshtein distance, in combination with the available ATC codes from the dataset. The results were assessed using variable similarity and distance thresholds, leading to high accuracy as the subset size increases. In the end, our algorithm was tested against a Linked Data approach that detects similar drugs while querying the DBpedia knowledge graph. The Linked Data approach was shown to perform slightly better as the data subset size increased, while it was outperformed by our proposed algorithm for small sized datasets.

Acknowledgment

The work in this paper was partially financed by the Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje.

References

[1] Dbpedia. Accessed 08 Apr 2019. http://dbpedia.org/.
[2] ATC Codes: Structure and Principles. Accessed 09 Apr 2019. http://www.whocc.no/atc/structure_and_principles.
[3] Global Open Drug Dataset. Accessed 12 Apr 2019. http://godd.finki.ukim.mk/.

[4] Active Substance. Accessed 13 Apr 2019. https://www.ema.europa.eu/en/glossary/active-substance.

[5] Paracetamol Brand Names. Accessed 13 Apr 2019. https://adf.org.au/drug-facts/paracetamol/.

[6] SPARQL. Accessed 13 Apr 2019. https://www.w3.org/TR/rdf-sparql-query/.

[7] Wiki Page Redirects. Accessed 13 Apr 2019. http://dbpedia.org/ontology/wikiPageRedirects.

[8] OpenRefine. Accessed 24 Apr 2019. http://openrefine.org/.

[9] A. Baftiu, C. Johannessen Landmark, V. Nikaj, I.-L. Neslein, S. I. Johannessen, and E. Perucca. Availability of Antiepileptic Drugs Across Europe. *Epilepsia*, 56(12):e191–e197, 2015.

[10] C. Brun and C. Hagege. Semantic Compatibility Checking for Automatic Correction and Discovery of Named Entities, Aug. 16 2011. US Patent 8,000,956.

[11] X. Chu, I. F. Ilyas, S. Krishnan, and J. Wang. Data Cleaning: Overview and Emerging Challenges. In *Proceedings of the 2016 International Conference on Management of Data*, pages 2201–2206. ACM, 2016.

[12] L. Ding and B. Dong. Language Independent Entity Linking. pages 724–729, 11 2018.

[13] S. Eltyeb and N. Salim. Chemical Named Entities Recognition: A Review on Approaches and Applications. *Journal of cheminformatics*, 6(1):17, 2014.

[14] S. García, J. Luengo, and F. Herrera. *Data Preprocessing in Data Mining*. Springer, 2015.

[15] W. H. Gomaa and A. A. Fahmy. A Survey of Text Similarity Approaches. *International Journal of Computer Applications*, 68(13):13–18, 2013.

[16] A. Hasnain, M. R. Kamdar, P. Hasapis, D. Zeginis, C. N. Warren, H. F. Deus, D. Ntalaperas, K. Tarabanis, M. Mehdi, and S. Decker. Linked Biomedical Dataspace: Lessons Learned Integrating Data for Drug Discovery. In *International Semantic Web Conference*, pages 114–130. Springer, 2014.

[17] A. Huang. Similarity Measures for Text Document Clustering. In *Proceedings of the Sixth New Zealand Computer Science Research Student Conference (NZCSRSC2008), Christchurch, New Zealand*, volume 4, pages 9–56, 2008.

[18] U. Irmak and R. Kraft. Scalable Semi-Structured Named Entity Detection, 2011. US Patent 8,073,877.

[19] M. Jovanovik, B. Najdenov, G. Strezoski, and D. Trajanov. Linked Open Data for Medical Institutions and Drug Availability Lists in Macedonia. In *New Trends in Database and Information Systems II*, pages 245–256. Springer, 2015.

[20] M. Jovanovik, B. Najdenov, and D. Trajanov. Linked Open Drug Data from the Health Insurance Fund of Macedonia. In *10th Conference for Informatics and Information Technology (CIIT)*, 2013.

[21] M. Jovanovik and D. Trajanov. Consolidating Drug Data on a Global Scale Using Linked Data. *Journal of Biomedical Semantics*, 8(1):3, 2017.

[22] K. Li, Y. Li, Y. Zhou, Z. Lv, and Y. Cao. Knowledge-Based Entity Detection and Disambiguation, 2017. US Patent 9,665,643.

[23] S. Liu, B. Tang, Q. Chen, and X. Wang. Drug Name Recognition: Approaches and Resources. *Information*, 6(4):790–810, 2015.

[24] D. Nadeau and S. Sekine. A Survey of Named Entity Recognition and Classification. *Lingvisticae Investigationes*, 30(1):3–26, 2007.

[25] J. D. Rennie and T. Jaakkola. Using Term Informativeness for Named Entity Detection. In *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 353–360. ACM, 2005.

[26] H. Wang, Q. Sun, A. Vitry, and T. A. Nguyen. Availability, Price, and Affordability of Selected Essential Medicines for Chronic Diseases in 11 Countries of the Asia Pacific Region: A Secondary Analysis. *Asia Pacific Journal of Public Health*, 29(4):268–277, 2017.

Practical evaluation on serious games in education

Slavica Mileva Eftimova
High School
SOU Jane Sandanski
Strumica, North Macedonia
meslavica@yahoo.com

Ana Madevska Bogdanova
Faculty of Computer Science and Engineering
Ss. Cyril and Methodius University
Skopje, North Macedonia
ana.madevska.bogdanova@finki.ukim.mk

Vladimir Trajkovik
Faculty of Computer Science and Engineering
Ss. Cyril and Methodius University
Skopje, North Macedonia
trvlado@finki.ukim.mk

Abstract

The arrival of the new learning methodologies is in response to the reality: new generations should learn in a different way. The so-called "Millennials" are looking for another kind of stimulus. Discussions for modernizing the curriculum include various solutions to retain students' attention and, in order to ensure that teachers learn how to act with a critical attitude, they will be confident and with the developed creative skills that they will need for success in the professional world in the future. The game based learning is more than providing educational games to students, it is about changing students' access to learning and their learning approach: the goal is to enjoy the learning process itself. This paper presents a methodological tool based on an evaluation framework for integration of digital games into education (MEDGE), expanded by adding additional information from the students, MEDGE+. The evaluation framework is used on three different approaches to the educational content: robot, micro: bit and playing quiz Kahoot. MEDGE+ provides better tool for the teachers in order to follow the student's interest when choosing appropriate educational games in the teaching process.

Keywords: serious games, critical attitude, games evaluation framework

I. Introduction

Today, the teacher abandons the role of a central figure, becomes a leader who guides the students through the learning process, enabling more learning styles, so that the student can move forward with his\her learning pace [1]. Students, on their part, use technology for communication, searching and finding information, expanding social experiences, and enjoying computer games on a daily basis [2]. The digital games (on their phones or computers) are played by students with a lot of energy and enthusiasm. This commitment is a challenge for the teacher - the learning process can be done through computer games.

Students have transferable skills to share online research and access to many digital texts in a number of contexts. If digital literacy is encouraged throughout the teaching program, using positive language is necessary. By making such changes in the language teachers use, with the goal of becoming closer to their students' language, the students themselves will feel closer to the teachers and will understand the learning material more easily [3].

Technology offers a wide range of opportunities for developing learning experiences across a wide range of topics. If digital literacy is promoted through the curriculum, a positive language is required. By changing the language that teachers use, in order to be closer to the students, then the students themselves will feel closer with the teachers and will easily overcome the material.

II. Background

With carefully selected concepts and accompanying pictures that create a pleasant and creative atmosphere, children learn more easily through various activities of games and tasks. The games often have a fantastic element that intrigue players and engage them in learning activities [4,5,6].

But in order to apply games in teaching, more conditions need to be met.

According to the UNESCO framework, teachers should use educational (serious computer games) in education, preferably in accordance with the application of the teaching experience [3]. Teachers should have the following competencies:

- Using the Internet for online research;
- Using tools for making text and spreadsheets, making presentations;
- Using communication and collaboration tools such as emails, video conferencing and social networks;
- Application of ICT sources for curriculum development;
- Interest in continuing upgrading and improving the teaching content they teach and their teaching skills;
- Knowledge of the subject they teach to be appropriate for the age of the students;
- Have managerial and organizational skills;
- Knowledge of strategies that will help the student to gain in-depth knowledge such as:

- Learning Collaboration;
- Problem-based learning;
- Project-based learning;
- Activities based on project development;
- Games and simulations;
- Research experiments;
- Case study;
- Exercises;
- Mentoring;
- Evaluation.

What conditions should the school have in order to introduce educational games in teaching? The answer to this question is divided in two parts:

- provided technical equipment in the classroom;
- teachers have to have appropriate digital competencies.

The word "competence" means knowledge or expertise in a given area [7]. Accordingly, digital competence is the ability to track, analyze, evaluate, generate and transmit information in digital format. This applies to desktops, laptops, smartphones and similar devices. Regarding the discussion of this term, there are various attempts to define definitions that are in use, as well as a few related names for it, such as information, the Internet or media competence. A person who is digitally competent will have more interlinked skills: knowledge of the basic principles of computer hardware, computer networking skills, the ability to engage in online communities and social networks. By digitizing human knowledge and developing digital technologies (mass production of devices that have access to the Internet). We can conclude that a digitally literate person will have practical knowledge of hardware and software, but also different kind of knowledge that they did not have in the last century. Computer literacy is often considered today as the ability to use the computer programs for some less complex practical tasks or the ability of individuals to effectively use the computer. Digital competence, or in other words, digital literacy, is considered to be as important today as reading and writing. Digital devices are starting to be used from an increasingly young age, but this does not help much young people to develop the skills they need for further personal and professional upgrading. Digital competence is far more than just accepting new technologies and using social media in order to create some content.

Today, digital literacy is almost equally necessary to attain personal and professional ambitions. It allows seemingly complex tasks to be performed in a much simpler and more efficient way and with better results. It is necessary to focus attention on the way the students use devices in extracting knowledge.

Digitalization and inroads have already led to major changes in our daily lives and our world in terms of information and work. However, these numerous changes have not yet become clearly visible and understandable to us. For this reason, it is particularly important to pay attention to large volumes of new information and to all innovations. To do this, it is necessary to have a sufficiently high level of digital competence.

As far as the technical equipment of the schools is concerned, as a first requirement is that there is an Internet connection and at least one laboratory with a certain number of computers, preferably connected in a local network. Possession of additional tools and equipment can greatly enrich hours with certain activities. The LEGO Mindstorm EV3 Robot [8] and micro:bit tools [9] are used in practical case studies.

III. The case study

In order to conduct this evaluation, the evaluation framework for integration of digital games into education (MEDGE) was used [10], expanded with two new questions, thus MEDGE+.

The following questions were answered by several professors at the “Jane Sandanski” High School:

- Is the game easy to use? (EASY)
- What is the educational goal of the game? (VAL)
- Does the game adapt to educational goals? (ADT)
- Pleasure / acceptance of the game by students? (QoE)
- What is the teacher's subjective opinion about the game? (SUB)
- What is the motivation of students to adopt the material? (MOT)

In order to achieve better motivation for the students when applying serious games, two questions were added, where the students were asked the following questions (MEDGE+):

- Through the game, I will easily overcome the material (EL);
- Through the game, my motivation for material adoption (MS) will increase.

The following scale of responses was offered:

I totally agree (5); I agree (4); I am neutral (3); I disagree (2); I completely disagree (1).

A. Case study 1 - Lego

A class in high school using a memory game (with the help of the LEGO Mindstorm EV3 Robot) [8].
Teaching unit: One-dimensional arrays
Type of lesson: lecturing

In order to get the students interested as much as possible on this thematic unit, a memory game in Python was used through simulation of robotic games. Preparing for this guide: LEGO Mindstorm EV3 Robot is the tool used for this game. A robot with specific parts is built in the instructions for building the LEGO Mindstorm Education Core Set.

The goal and the game is to build the main body for the robot (base unit) and the color sensor.

Effects

Computer Science (Python) - This lesson will help students understand the use of an array, from an abstract concept to a point where they actually understand how the color storage works in random order. Students will be introduced to the random functions used to generate colors.

Exercise

Create a program that will put random colors in sequence, and then the robot will repeat (express) the order of colors. The student must remember the colors and show colors on the card to the robot sensor in the same order as given. In the end, the robot announces whether the study of time wins or not.

Reflection

Students learned how to store colors in arrays and to check if the generated colors are the same as shown in front of the sensor. They can also make different versions of this program by counting a score and saving other type of data.

Based on a poll conducted after playing this game, the following results were obtained, given in the Fig1:

	EASY	**VAL**	**ADT**	**QoE**	**SUB**	**MOT**	**EL**	**MS**
robot	2,5	3,4	5	3,8	4,4	4,6	**4,4**	**4,1**

Fig. 1. Review of robot responses, MEDGE+

B. Case study 2 - Kahoot

Teaching unit: Basics in programming with C++; *Type of lesson*: Kahoot quiz to check the acquired knowledge on the topic Introduction to programming in C ++.

Kahoot [11] is a formative learning tool that uses quizzing technology, discussions and surveys. The principle of work is basically a game in which the whole class participates in real time. For the preparation of this class, a quiz with 10 questions was developed, which examines the gained initial knowledge in programming in C ++, which is necessary to start programming the simplest tasks in C ++.

Exercise Students get the link and join the game.

Reflection

The class with Kahoot was filled with excitement and euphoria like no previous one. The competitive spirit was at the highest level. After the quiz was conducted, questions that were incorrectly answered were discussed. The students asked after each teaching unit (or at least after completing a theme) to have a time dedicated to competing with the Kahoot Quiz**.** Based on the conducted survey after playing this game, the following results were obtained, given in the Fig2:

	EASY	**VAL**	**ADT**	**QoE**	**SUB**	**MOT**	**EL**	**MS**
Kahoot	4,6	5	5	5	4,7	5	**4,8**	**4,7**

Fig 2. Review of Kahoot responces, MEDGE+

C. Case study 3 – micro:bit

A class in high school using a micro:bit [9] to verify acquired knowledge of algorithms with a branched structure.

Teaching unit: Algorithms and programming;

Type of lesson: Algorithms and their representation. In order to perform this lesson, primary school Sando Masev Strumica was visited. This school owns 30 micro:bit devices [12], obtained with the help of British Council. At this class, the application of the algorithm with a branched structure was presented, realized practically with the micro:bit device.

Exercise

The acquired experiences were used for introduction into the branch structure. Other approaches were introduced to explain algorithms with a branch structure. The students were able to solve other examples.

Reflection

By using the micro:bit, the programming becomes clearer, closer and more attentive to the students.

The survey by MEDGE+ has produced the table in Fig 3.

	EASY	**VAL**	**ADT**	**QoE**	**SUB**	**MOT**	**EL**	**MS**
micro:bit	4,3	5	5	4,5	4,2	5	**4,8**	**4,5**

Fig.3. Review of micro: bit responces, MEDGE+

The evaluation methodology MEDGE+ produces the net presented in Fig.4. It gives graphical representation of the game/tool acceptance in the three case studies.

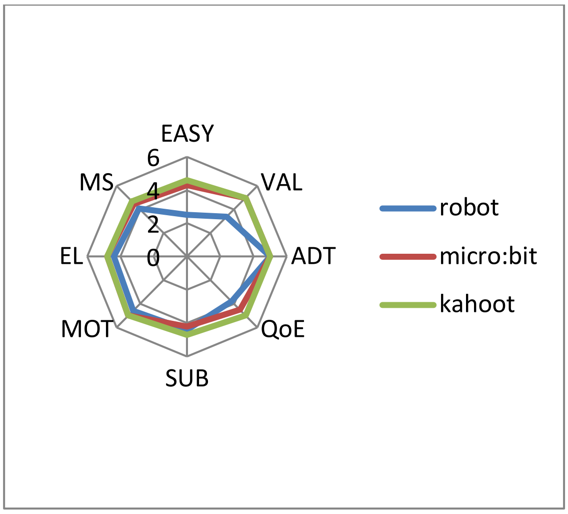


Fig4. Evaluation of the explored games with MEDGE+

IV. Discusion

In the following discussion, all of the case studies are elaborated by reviewing the positive and negative aspects of the serious games in the learning process. At the lesson of one-dimensional strings using the robot, the positive characteristics were: greater interest and curiosity towards the innovative approach of teaching resulting in increased interest in learning and interest in research using additional sources for the new concepts. The negative aspect is that the use of robots in teaching requires additional budget. Furthermore one robot is a not enough for a group of 24-30 students. Another point is the programming language - in order to use the robot in the teaching process, a different programming language from the one studied in the regular classes was needed. The good sides in the realization of classes using the Kahoot quiz for repeating the material for the basics of programming were: initiating a competitive spirit, raising awareness of teamwork, getting quick results for the correct answers and showing greater interest in understanding the reasons of choosing the wrong answers. Also, all students answered the same questions at the same time and there was no fear of consequences if wrong answer was submitted.

In the third case study, the algorithms for a branched structure with micro: bit was introduced. The positive side was this new approach to learning algorithms. It was more interesting to the students because of the visualization and the ease of use. One disadvantage of using micro:bit in high schools is that following the curriculum, micro:bit can be applied only at the beginning of the programming courses because the latter material is more complex.

V. Conclusion

The introduction of games in IT teaching is a very positive experience [13,14,15], but of course, the realization of each of these serious games and tools has positive and negative aspects.

In this paper, following the evaluation methodology [10], MEDGE+ was introduced that can give even more insight when choosing an appropriate educational game in the teaching process. We have explored three case studies, and used the MEDGE+ methodology to measure the acceptance of the given games and tools.

The performed analysis was done over three different games: robot simulation, Kahoot and micro:bit coding. It showed that all three activities positively influenced the process of adopting new knowledge and knowledge testing, while the greatest satisfaction and motivation according MEDGE+ achieved the quiz Kahoot. The extension of the methodology [10] considered the inclusion of the student's opinion, contributing to the teacher's decision which games/tools should be chosen in the educational process.

The use of games in teaching obviously has many advantages, and as a future work, more examples of games and tools will be evaluated using the information provided by MEDGE+.

References

[1] L. Phipps, V. Alvarez, S. de Freitas, K. Wong, M. Baker, and J. Pettit, "Conserv-AR: A Virtual and Augmented Reality Mobile Game to Enhance Students' Awareness of Wildlife Conservation in Western Australia", Proceedings of the 15th World Conference on Mobile and Contextual Learning (mLearn 2016), Sydney, Australia, vol. 1, pp. 214-217, 2016

[2] H.A. Spires, "21st century skills and serious games: Preparing the N generation," in L.A. Annetta, Serious educational games. Rotterdam, The Netherlands: Sense Publishing, 2008

[3] United Nations Educational, Scientific and Cultural Organization, unesco ict competency framework for teachers, 2011;

[4] P. Fotaris, T. Mastoras, R. Leinfellner, and Y. Rosunally, "Climbing Up the Leaderboard; An Empirical Study of Applying Gamification Techniques to a Computer Programming Class," Electronic Journal of e-Learning, vol. 14, no. 2, pp. 94-110, 2016.

[5] T.M. Connolly, E.A. Boyle, E. MacArthur, T. Hainey, and J.M. Boyle, "A systematic literature review of empirical evidence on computer games and serious games," Computers & Education, vol. 59, pp. 661–686, 2012.

[6] J.C. Burguillo, "Using game theory and Competition-based learning to stimulate student motivation and performance," Computers & Education, vol. 55, no. 2, pp. 566–575, 2010

[7] Digital competence https://europass.cedefop.europa.eu/resources/digital-competences , 15.12.2018;

[8] Lego education, https://education.lego.com, 25.02.2019;

[9] micro:bit, https://microbit.org/code/, 25.02.2019;

[10] Maja Videnovik , Ana Madevska Bogdanova , Vladimir Trajkovik, "Serious games evaluation methodology", Proceedings of ICERI2018 Conference 12th-14th November 2018, Seville, Spain;

[11] M. Videnovik, L. Kionig, T. Vold, and V. Trajkovik, "Testing framework for investigating learning outcome from quiz game: A Study From Macedonia and Norway," in 17th International Conference on Information Technology Based Higher Education and Training (ITHET), pp. 1-5, IEEE, 2018

[12] materials by teacher Biljana Nikolova, OO Sando Masev Strumica, 15.12.2018;

[13] Gamestorming, https://gamestorming.com/category/games-for-problem-solving/, 12.02.2019;

[14] M. Popescu, S. Arnab, R. Berta, J. Earp, S. De Freitas, M. Romero, I. Stanescu, and M. Usart, "Serious games in formal education: discussing some critical aspects," in Proceedings of 5th European Conference on Games-Based Learning, Athens, Greece, pp. 486–493, 2011

[15] Immersing, addaptive learning https://www.ixl.com, 12.02.2019

Predictive Policing in City of Skopje using ArcGIS

Elena M. Jovanovska, Dimitar Kovachevich, Andreja Naumoski
Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje
Skopje, Republic of North Macedonia
e-mail: jovanovska.elena14@gmail.com, dimitar.kovachevich@students.finki.ukim.mk, andreja.naumoski@finki.ukim.mk

***Abstract* — The desire to predict unwanted events before they happen has always been present in society, especially when the same events would bring in question a person's life either welfare. Predictive analytics is considered advanced and used for prediction of unknown events. The security services in developed countries use this type of analytics and prediction of criminal cases called Predictive Policing. With the expose of online public information and more sophisticated tools for modeling and analysis, where individuals or companies unrelated to the state can create specialized software for this purpose.**

Keywords — Crime, GIS, Analytics, Prediction, Predictive policing.

I. Introduction

With the continuous increase of population and town's density, the police and other security services face difficulties in providing law and order and adequate response to treads. The continuous advance in technology and science enables the idea for prediction of unwanted events to become a reality and the methods used for predictive analysis are used more often to prevent these events.

A. Related Work

One research [1] deals with potential path areas because they incorporate both spatial and temporal data, time budget and mobility constraints making the results more accurate. They approach mapping activities of criminal cases in two ways like time-geographic density estimation in order to calculate individual activity spaces using potential path areas that have associated probabilities and secondly activity spaces of numerous individuals that are combined in a single intensity surface that maps areas of a city that are more frequented by offenders and higher crime expectancy like registered sex offenders and sex crimes in the city of St. Louis.

Another research [2] deals with identifying patterns or spatial concentrations called "hotspots" and acting upon the problem known as "policing". In other words, taking action before the problem occurs. To prevent crimes, police presence should be increased in the hotspot areas. Spatial analysis and crime mapping became popular in the law enforcement and scholar circles because of the new approach to prevent crime and the increasingly cheaper and easy-to-get data collections. The location and the time of the crimes is something the police is systematically collecting, making the creation of hotspots and prediction possible.

The conceptualization of predictive policing, its potentials and realized benefits as well as drawbacks are discussed in the literature review [3] that shows their discrepancy and the available empirical evidence. Some empirical evidence provides little support for the claimed benefits, whereas other studies conclude there was a decrease in crime, and some results are neutral. Furthermore, there is no evidence concerning the drawbacks of the approach. This realization encourages researches to do independent tests with both negative and positive outcomes in order to generate real evidence base for predictive policing.

When crimes happen, often social networks can serve as a ground for providing additional information in the big data collection. Such can be the case with Twitter and its posts associated with disorder. The research [4] estimates the utility of social media to explain variance in offline crime patterns, provides first evidence of the estimation using a measure of broken windows found in the textual context of social media communications, it tests if there is a difference in offline and online communications, it takes the results of experiments to participate in debates on big data and data prediction

B. PredPol Popular Framework

One popular platform that emerged out of a research project between LAPD and UCLA is the PredPol predictive policing software [5]. It all began when the chief of the department wanted to use COMPSTAT data for more useful purposes like providing forward-looking recommendations concerning when and where crimes could occur. In cooperation with UCLA and Santa Clara University where mathematicians and behavioral scientists determined a variety of data types and forecasting models, which were later refined with crime analysts, three main points were determined like crime type, location, date and time. The popular platform was created through 70 research years of analysis, modeling, development and constant testing. Today, it is serving as a precise definition for predictive policing in identifying times and locations where specific crimes could happen and trying to prevent victimization. The information used is anonymized because the civil rights for privacy of the residents should be protected.

II. Materials and Methods

The goal of this project is to pin the hotspots or areas with high activities for different crimes, but in our case, it is criminal activities and offenses within the city of Skopje as well as analysis of their connection with several factors like population density, percentage of poverty, income and etc.

Publicly available criminal records from the Ministry for Internal Affairs website contain data for the total number of certain type of crimes or offenses, grouped by city region and years and/or months. Thus, to perform a detailed analysis is hard, because data is gathered for the whole city of Skopje and not divided by municipalities. Meaning that the exact locations of past crimes are unavailable.

The same case applies for data that can be used to analyze the correlation of the offenses such as salary, population density, social status… As a result, at this stage of development we used the data from the Ministry of Internal Affairs for most frequent offenses of public law and order, combined with few thousand locations from other sources whose coordinates were transformed to fit the area of Skopje.

The source of data for this project is taken from the publicly available criminal records on the Ministry for Internal Affairs website and imported into Excel sheet, in which there are almost 4700 rows with offenses of public law and order from 10 amendments in the time period from 2012 to 2017, with a separate Sheet for every year and one sheet with all years incorporated. Every year has approximately 780 rows of data and the type of offenses are accordingly divided upon the percentage of incidence in a year time period. The map of Macedonia is free and taken from geofabrik.de.

The coordinates for every row are taken from crimes that happened in Somerset and Avon, Great Britain (GB), where the existing latitude and longitude were modified to fit the territory of Skopje, but their distribution is almost unchanged. Initially the data rows from GB were 12000. After the coordinate transformation, a large part of the locations was not found on the territory of Skopje. These locations were removed with the help of Select by Location tool, where the method for selection, the source layer and the appropriate method were chosen (Fig.1).

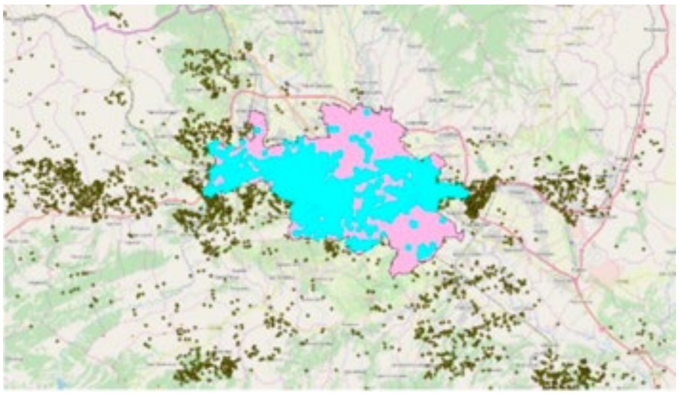


Fig. 1. Representation of the entire dataset on the map of city of Skopje.

Furthermore, estimations were performed to get the percentage for every type of offense by year and accordingly to fill the table with the following attributes: CID (offense identifier), Year (year of reported offense), Reported_by (station where the offense was reported), Falls_within (station that solves the case and takes over the investigation), Longitude, Latitude, Crime (offense type).

Every Sheet from the table is inserted in an appropriate geodatabase table in the project with right click (Export > To Geodatabase (multiple)). With right click on the tables > Display XY data and the appropriate selection of attributes over which the coordinates will be parsed, the offenses are shown on the map. For a coordinate system we chose the geographical coordinate system WS 1984, because it is necessary to have a geographical instead of a projected coordinate system in order to use the latitude and the longitude.

The layer presented on Fig.2, show offenses by year are created using the predefined settings, and shows the total number of offenses from 2012 to 2017, which takes the type of offense for attribute (Crime) and the imported map. At this point, data outside Skopje City boundaries are deleted.

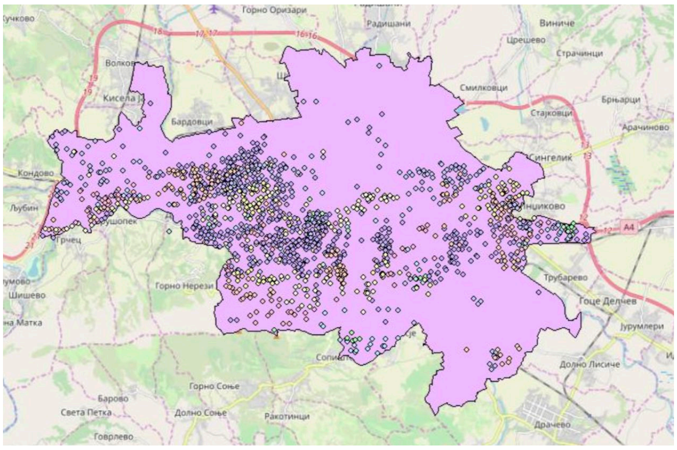
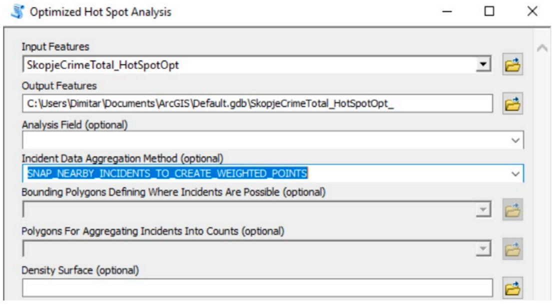


Fig. 2. Representation of crimes that happened in Skopje from 2012-2017 organized by type.

III. GIS Modeling and Results

After the settings in Geoprocessing > Environment Settings are adjusted, more specifically the cartographic system for the layer of Skopje city and the mask for Raster Analysis to be the same layer in order to get an appropriate presentation of the city borders interpolation, the Hotspot analysis can be executed. For the analysis Optimized Hot Spot Analysis is used with the settings presented in Fig. 3.



Optimized Hot Spot Analysis

Input Features
SkopjeCrimeTotal_HotSpotOpt

Output Features
C:\Users\Dimitar\Documents\ArcGIS\Default.gdb\SkopjeCrimeTotal_HotSpotOpt_

Analysis Field (optional)

Incident Data Aggregation Method (optional)
SNAP_NEARBY_INCIDENTS_TO_CREATE_WEIGHTED_POINTS

Bounding Polygons Defining Where Incidents Are Possible (optional)

Polygons For Aggregating Incidents Into Counts (optional)

Density Surface (optional)

Fig. 3. Optimized Hot Spot Analysis settings panel.

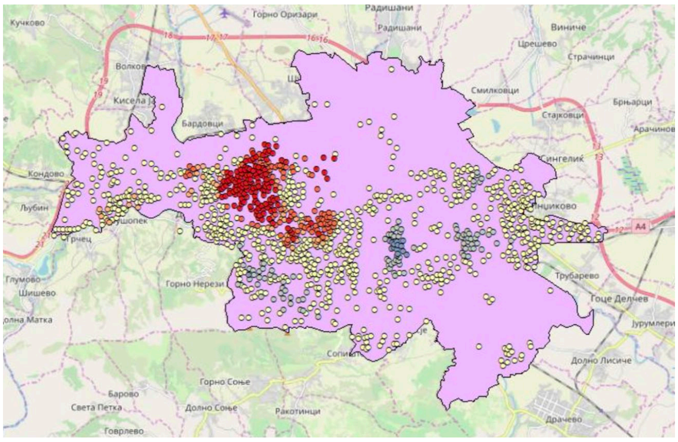


Fig. 4. Hot spots for the appropriate layers for total number of crimes in the time period 2012-2017.

The hotspot analysis depicts easy to see hotspots where most of the crimes happened. They are concentrated around the city centre and the west from the city centre. In order to get a better view of the areas with increased number of offenses, interpolation is used (Spatial Analyst Tools > IDW). Furthermore, the layer got from the hot spot analysis is used as input, and for data the points from the same layer (Gi_Bin). After the execution, we get the following layer presented with Fig. 5. Again, as in Fig, 4, from the result it is easy to see the hotspot where most of the crimes are concentrated, around the city centre and the west from the city centre.

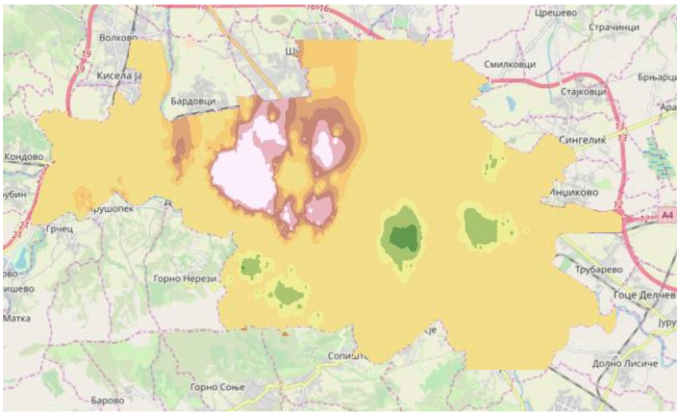


Fig. 5. Interpolation from the Hotspot analysis for the total number of offenses from 2012-2017.

Furthermore, we analyze the frequency of the accidents by type of incident or by year, in our case for the year of 2016, as it is done in Fig. 6. The presented map illustrates almost the same distribution of crime hotspots, but Fig.6 depicts higher level of crime towards south part of the previous hotspot found in Fig. 5.

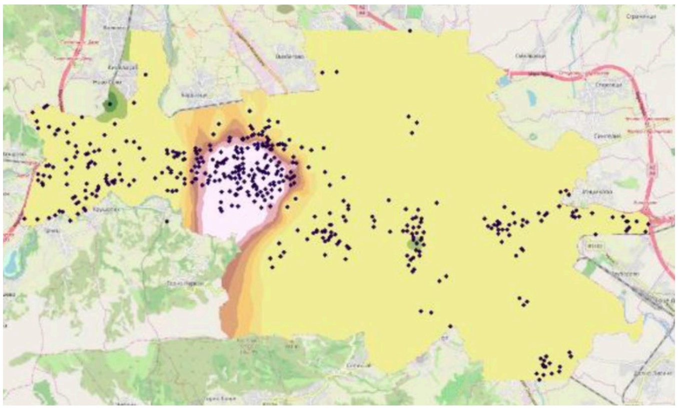


Fig. 6. Interpolation of the Hotspot analysis for the total number of offenses for 2016.

IV. Conclusion

Through the development of this project, practical application for the analysis of offenses and criminal activities for a selected area was shown. With the available data for different factors where analysis for their connection with the offenses can be performed, we can get more precise results and this project can become applicable in a real-life scenario. The tools [6] for predictive policing rising from the emerging science combine crime mapping, statistical analysis and law enforcement expertise to perform time and place crime prediction. This approach may increase police effectiveness, reducing cost and time needed for help to arrive.

Acknowledgment

This work was partially financed by the Faculty of Computer Science and Engineering at the Ss. Cyril and Methodius University in Skopje.

References

[1] Downs, J. A. “Mapping sex offender activity spaces relative to crime using time-geographic method,” Annals of GIS, vol. 22, no. 2, pp. 141–150. 2016.

[2] Townsley, M. “Crime mapping and spatial analysis,” In B. Leclerc & E. Savona (Eds.), Crime prevention in the 21st century. Cham, Switzerland: Springer International Publishing, pp. 101-112, 2017.

[3] Meijer, A., and Wessels, M., “Predictive Policing: Review of Benefits and Drawbacks,” International Journal of Public Administration, 2019, pp. 1-9.

[4] Williams, M. L., Burnap, P., and Sloan, L. “Crime sensing with big data: The affordances and limitations of using open source communications to estimate crime patterns,” British Journal of Criminology, vol. 57, pp. 320–340, 2016.

[5] https://www.predpol.com, Accessed 12 March 2019.

[6]https://www.dhs.gov/sites/default/files/publications/GIS-Predictive-Policing-AppN_0813-508_0.pdf, Accessed 23 March 2019.

STUDENT PAPERS

A Brief Introduction to Persistent Homology

Petar Sekuloski
Faculty of Computer Science and Engineering
Ss. Cyril and Methodius University, Skopje
petar.sekuloski@finki.ukim.mk

***Abstract*—In this paper, it is presented a brief introduction to Persistent Homology. With the introduction, we will illustrate how some topological characteristics can be interesting for data analysis problems and give a short introduction to Topological Data Analysis. Also, we will present some results of applying Persistent Homology to some real world data.**

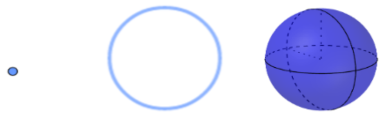
***Keywords*—topological data analysis, data science, machine learning, algebraic topology**

I. Introduction

Topology is mathematical field that studies properties of topological spaces that are invariant of continuous deformations such as connectedness and compactness. In algebraic topology, homology gives a general way of associating algebraic structures such as abelian groups to topological spaces. We can say that homology is a set of topological invariants of some topological space which is represented by its homology groups. These invariants gives us an information about connected components and holes of the topological space. Homology groups are associated with different dimensional structures of the topological space which is observed. The number of structures for some dimension k is the rank of the k-dimensional homology group of the topological space. That number is also called Betti number, β_k , of a dimension k. See *Figure 1*, for Betti numbers of commonly studied low dimensional topological spaces. For a higher dimensional spaces, these numbers gives the k-dimensional holes.

The application of homology and other algebraic topology concepts to data analysis is main idea in Topological Data Analysis. One of the Topological Data Analysis's techniques is Persistent Homology. We will illustrate how works Persistent Homology with an explanation of the necessary mathematical concepts and we will discuss some results from applying Persistent Homology on Statlog (Heart) dataset.

Since Topological Data Analysis is not wieldy known filed to computer science students, computer scientists or data scientist and it is included in most of Data Science, Machine Learning or Computer Science academic programs, the main goal of this paper is to provide a brief introduction of Topological Data Analysis. Also, there will be mentioned some successful applications to some real world Machine Learning and data analysis problems.



β_0	1	1	1
β_1	0	1	0
β_2	0	0	1

Figure 1. A k-th Betti number β_k for a point, circle and sphere. For k=0 it measures connectivity, k=1 it measures loops and for k=2 it measures voids.

II. Simplicial complexes

Definition 1. For a given set $S = \{x_0, x_1, , \dots, x_k\} \subseteq \mathbb{R}^d$, a convex combination of a points in S is a point of the form $x=\sum_{i=0}^{k} \lambda_i x_i$, where $\sum_{i=0}^{k} \lambda_i = 1$ and $\lambda_i \geq 0$.

A convex combination is a linear combination which is affine combination with non-negative scalars.

Definition 2. The set of all convex combinations for a given set of points S is called convex hull.

Definition 3. A convex hull of $k+1$ affinely independent points $S = \{x_0, x_1, , \dots, x_k\} \subseteq \mathbb{R}^d$ is called a simplex. The points of S are called vertices of the simplex.

A k-simplex can also be denoted as σ^k. σ^k is a k-dimensional subspace of $\mathbb{R}^d$. Also, we can say that the dimension of σ^k is k.

The low dimensional simplices (plural: simplices or simplexes) have special names:

- a 0-simplex is called a *vertex;*
- a 1-simplex is called *edge;*
- a 2-simplex is called *triangle*;
- a 3-simplex is called *tetrahedron;*

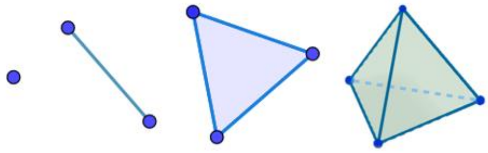


Figure 2. 0-simplex 1-simplex, 2-simplex, 3-simplex

Definition 4. A family Δ of non-empty finite subsets of a set S is called a simplicial complex, if

1. For $\forall x \in S, \{x\} \in \Delta$.

2. If $\tau \subseteq \sigma \in \Delta$, then $\tau \in \Delta$.

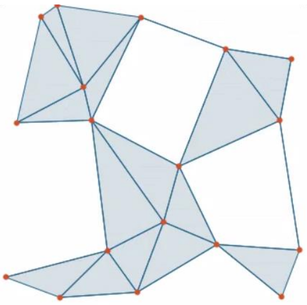


Figure 3 An example of simplicial complex

We call the sets {x} the vertices of Δ. Last definition gives an abstract definition of simplicial complex that can be applied to a data where vertices will be some data points. Topological invariants of the space, such as: holes and number of connected components, can be computed from a simplicial complex, see Figure 3. One of the key idea of Topological Data Analysis is to construct a simplicial complex from a dataset. There a few ways to construct such as simplicial complex [1].

Persistent Homology is a method for computing a topological features of a space. That space should be represented by a simplicial complex. The problem here is that if we add just a single point in some space, topological invariants will be different . Persistent Homology provides a way to see which of the topological features are invariant, with computing features for a sequence of spaces, not just for a single space. For example, if we have points in R^2 that represent a complete graph where the edges represent a distances, we can choose how to connect points using only distance. The number of connected components increases when we connect two points, when the distance of one to another point is decreasing. Also, when we are working with higher dimensional space, topological features can show up and disappear for each part of the sequence of spaces. The most unchangeable features, stable features, computed for each part of the sequence of spaces are the invariants of the exact topological space that is reconstructed of that points [10]. Result of Persistent Homology is a topological summary named Persistent Barcode.

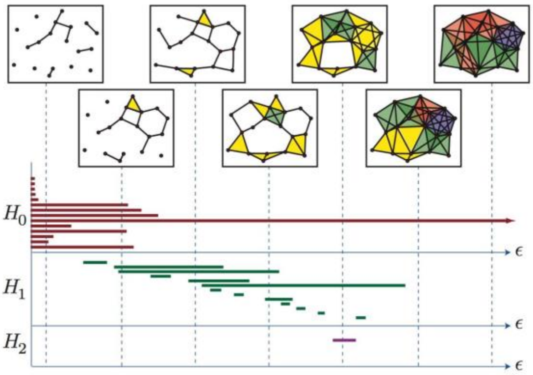


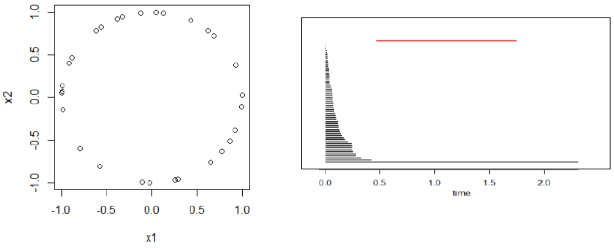
Figure 4. Illustrating Persistent Diagram https://www.math.upenn.edu/~ghrist/preprints/barcodes.pdf

III. Experiments

A. Computing Barcode for synthetic dataset

We generate a synthetic dataset which vectors represent points of a circle. Then we apply Persistent Homology and we generate a Barcode. The Black lines show how the points and the components merges. The Red line shows 1-dimensional hole.

Figure 5. For a given data that represents points (left) as input, we



compute a barcode.

From the computed Barcode we can see that red line exists at most of the time. That tells that when we are computing a homology for each space in the sequence of topological spaces, most of the topological spaces has 1-dimensional holes. So we can conclude that the original topological space has one 1-dimensional hole, or we can say that the topological space has one loop.

B. Analysis of Statlog (Heart) Dataset

In this experiment we tend to find which attributes of the dataset are most valuable using persistent homology. Pre-processing of data is not explained in these work, because the main purpose here is to see how Persistent Homology can be applied on Heart Statlog Dataset [6] and with these application to illustrate how this technique works. Some of the attributes are well known medical parameters as chest pain, blood pressure, maximum heart rate, fasting blood sugar, maximum heart rate achieved and etc. Also, there are sex and age as attributes. In other words we like to see which of the attributes affects the topological characteristic of the space at most. We will purpose a method for application of Persistent Homology, Figure 6.

Computing Barcodes
Compare results
Discus the results

Figure 6. Method for application of Persistent Homology

The first step of the method is to compute Persistent Homology. We are computing persistent homology. As a result of the computation we obtain the barcodes for each set of attributes as an input illustrated in Figure 7. Firstly, we set all attributes as an input, and we get the Barcode 0. This barcode gives us the topological characteristics of the whole dataset. For other computations we set the dataset with one attribute reduced in each step, as an input. Then we apply Persistent Homology on the inputs and we are computing the appropriate Barcodes.

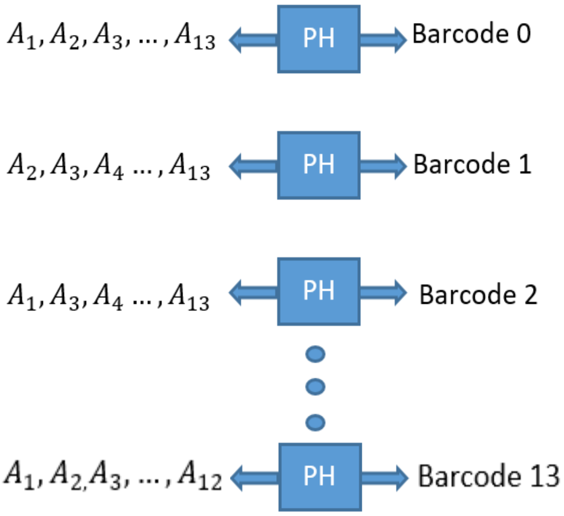

$A_1, A_2, A_3, \dots, A_{13}$
PH
Barcode 0
$A_2, A_3, A_4 \dots, A_{13}$
PH
Barcode 1
$A_1, A_3, A_4 \dots, A_{13}$
PH
Barcode 2
$A_1, A_2, A_3, \dots, A_{12}$
PH
Barcode 13

Figure 7. Computing Barcodes: A_i –i^{th}attribute of the dataset, B_i –i^{th} computed barcode for appropriate input.

The second step is to compare the obtained barcodes $B_1, B_2, B_3, \dots B_{13}$ with the barcode for whole dataset B_0. The goal is to find which of the barcodes $B_1, B_2, B_3, \dots B_{13}$ are different than B_0. This will give us which attribute or attributes affects the topological characteristics of the space. If some of the Barcodes are different that B_0, that means these attributes are most valuable for reconstructing the original space.

From the results of this experiment we can say that structural heart defect and number of major vessels colored by fluoroscopy [9] most affect the topological characteristics of the space. The results can not be visualized easily because we are not working with low dimensional space.

IV. Discussion

Determining the topological features of the space can be essential for further analysis of data. Applying this on real world data can give us the most valuable attributes of some dataset. Over the last few years, application of techniques of topological data analysis in machine learning and data science is rapidly increasing and a big progress is made. Interesting connection is application of Topological Data Analysis to deep learning which is a powerful method for analyzing complex data [8]. Many theoretical and practical issues should be concretized and it is an area that brings many topics for further research. Theoretical questions that are arising today are concerned with multi-dimensional persistence. There is significant progress in notion of multi-dimensional persistence but there are many computational obstacles which derives many problems for solving.

There are some successful applications of Topological Data Analysis in the following areas: graph reconstruction, complex networks, progression analysis of disease, image analysis, molecular biology [11]. This list is a short summary of application of TDA.

References

[1] H. Edelsbrunner, "Persistent homology: theory and practice", 2014.

[2] Gunnar Carlsson, "Topology and data". Bulletin of the American Mathematical Society. 46 (2), 2009, pp. 255–308.

[3] J. R. Munkres, Topology. vol. 2. Upper Saddle River: Prentice Hall, 2000

[4] llen Hatcher, Algebraic topology. Cambridge University Press, 2002

[5] G. Carlsson, A. Zomorodian, A. Collins, L. Guibas, J. (2005-12-01). "Persistence barcodes for shapes". International Journal of Shape Modeling.

[6] http://archive.ics.uci.edu/ml/datasets/statlog+(heart)

[7] Brittany Terese Fasy, Jisu Kim, Fabrizio Lecci, and Clément Maria. "Introduction to the r package tda. ", arXiv preprint arXiv:1411.1830, 2014.

[8] https://s3.amazonaws.com/cdn.ayasdi.com/wp-content/uploads/2018/11/12131418/TDA-Based-Approaches-to-Deep-Learning.pdf

[9] https://en.wikipedia.org/wiki/Fluoroscopy

[10] Ulrich Bauer and Michael Lesnick." Induced matchings of barcodes and the algebraic stability of persistence. In Proceedings of the thirtieth annual symposium on Computational geometry", p. 355, 2014.

[11] https://en.wikipedia.org/wiki/Topological_data_analysis

Air Pollution Prediction Using LSTM Neural Networks

Bojan Evkoski, Zafir Stojanovski, Aleksandar Trajkovski and Dejan Gjorgjevikj
Ss. Cyril and Methodius University
Faculty of Computer Science and Engineering
Rugjer Boskovikj 16, 1000 Skopje, North Macedonia
Email: {bojan.evkoski, zafir.stojanovski, aleksandar.trajkovski}@students.finki.ukim.mk
dejan.gjorgjevikj@finki.ukim.mk

***Abstract*—Air pollution in North Macedonia is 20 times over the EU limit. Recently Skopje is mentioned as the most polluted city in Europe. As a result, this is believed to contribute to 2000 annual premature deaths in Skopje, Tetovo and Bitola only. Being able to forecast air pollution levels to take timely precaution could drastically reduce these numbers. Using state of the art recurrent neural networks known as LSTMs, we were able to predict these levels by combining historical pollution data and weather forecasts through meta models, achieving mean RMSE for all sensors around 20, with the best results having RMSE as low as 8.78, with PM_{10} measurements ranging from 0 to above 1000 and are usually accompanied by a lot of noise. In this paper we present several approaches we have tried for solving the problem and a basic comparison between them and we also propose a way to expand these models into a real-time system for multitarget predictions.**

Keywords*— *air pollution forecast; LSTM & Meta models; univariate vs. multivariate comparison

I. Introduction

In the late 70's and early 80's, researchers dismissed initial studies indicating that air pollution is directly associated to daily mortality rates [1] due to the fact that these experiments didn't account for cigarette smoking and other health related issues. However, in 1993 a 14-years long U.S. study emerged [2] providing results that despite the inability to dismiss the effects of other unmeasured risk factors with certainty, fine-particulate air pollution does in fact contribute to excess mortality. These particles (aerodynamic diameter less than 2.5 micromillimeters) are thought to pose a particularly great risk to health because they are more likely to be toxic than larger particles and can be inhaled deeper into the lungs.

Recent data released by the World Health Organization [3] show that both prenatal and postnatal exposure to air pollution can negatively influence neurodevelopment, lead to lower cognitive test outcomes and influence the development of behavioral disorders such as autism spectrum disorders and attention deficit hyperactivity disorder. More recently, the Institute of Public Health of Republic of North Macedonia [4] showed that 1,903 human lives (excess deaths) are lost annually due to $PM_{2.5}$ exposures (22.3% of total all-cause (natural) mortality). If the limit values of the $PM_{2.5}$ particles had complied with the existing EU and WHO limit values, 908 lives could have possibly been saved, and 1547 respectively.

Being aware of the possible hazards air pollution imposes, our goal is to explore the ways of efficiently forecasting the pollution levels of these fine particles. In this paper we focus on predicting PM_{10} particles for reasons connected to availability and abundance of data, but since they are highly correlated with the $PM_{2.5}$, that should not be a problem. Ultimately, the mission is to provide a tool that will allow the citizens of Skopje to be in control of how exposed they are to ambient air pollution.

The remainder of the paper is organized as follows: in section II we describe the theoretical background of LSTMs and their advantage to standard RNNs, while section III presents our data and the preprocessing stage. In Section IV we make an overview of the models we are using for prediction and finally in section V and section VI we compare the results, share future plans and conclude the paper.

II. Theoretical Background

In this paper, we propose using an RNN called long short-term memory (LSTM), to analyze time series air pollution in Skopje. The LSTMs take as input not only the current timestep input, but also what they have "perceived" previously in time. While RNNs (Recurrent Neural Networks) are suitable fit for modeling time series data, they have been known to have a serious issue - vanishing gradient [5]. This occurs as result of the many multiplications that occur within the hidden layers of the net, creating derivatives that vanish as they progress through the network while backpropagating. In simple terms, the network learns incredibly slow, or in some cases does not learn at all. On the other hand, LSTMs solve this problem by preserving the error in a gated cell, thus the gradient is calculated very differently from the standard RNN [6]. Since the LSTMs already demonstrated their usability for time series prediction in recent years, the decision to use this algorithm was unequivocal. The comparison between a simple RNN and an LSTM RNN is shown in Figures 1 and 2, respectively.

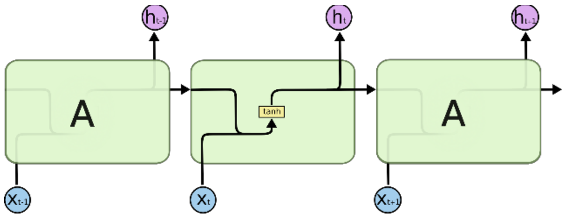


Fig. 1. RNN with one layer and no gated memory cells

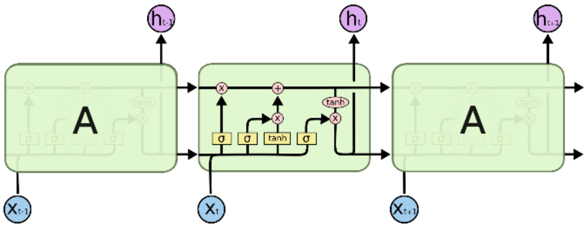


Fig. 2. LSTM RNN with gated memory and sigmoid activation functions

III. DATA

A. *Description*

The data were obtained from two separate sources. First, we queried air pollution data from Skopje Pulse (https://skopjepulse.mk/) generated by six sensors owned by the government located in the following locations around Skopje: Centar, Lisiche, Miladinovci, Karposh, Gazi Baba and Rektorat. Each of these sensors were put into use at different times, with the oldest one being online for 14 years, dating back to 2005. Next, we queried the data from Dark Sky's API (https://darksky.net), acquiring information about Skopje's hourly weather from 2012 until 2019.

B. *Preprocessing*

The preprocessing was done using the Pandas toolkit in Python [7]. Both the air pollution and the weather datasets were hourly, each having timestamp as an index column. The air pollution dataset had a single feature: value of the measured PM_{10} concentration. On the other hand, the weather dataset had the following features: cloud cover, dew point, humidity, temperature, UV index, visibility, wind speed and wind direction. The wind direction attribute contained categorical values such as the following: N (North), NE (North-East), NNE (North-North-East) etc. Instead of simply enumerating the values, we decided to represent this attribute as two separate ones – Wind X and Wind Y, expressing the spatial proximity between the values. If we were to enumerate them, 1 being N and 16 being NNW – we can immediately see the issue: naturally these values are extremely near, but their enumeration values are the furthest apart. Thus, given the spatial arrangement of the wind values, we obtained the X and Y components by appropriately taking the cosine and sine of the angle being formed. Fig.3 shows the circle of wind directions and their sine and cosine mappings.

Finally, we augmented the weather dataset by adding two additional attributes: a boolean indicating if it's a workday or not, and the month. Again, for reasons described above, we transformed the month value as two separate attributes (ex. January is as close to February as it is to December). From now on, we refer to this augmented dataset as *extras*.

C. *Data Partitioning*

Based on the model architectures described below, we split the datasets of all the different sensors in the following manner:

- TRAIN1 – for training the LSTM (from 2005 until 2015)
- TRAIN2 – for training the SVR meta model (from 2015 until 2017)
- TEST – for testing the both LSTM and meta model (only 2018)

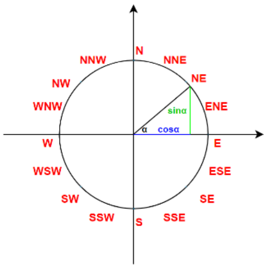


Fig. 3. Wind direction with 2D angle mappings

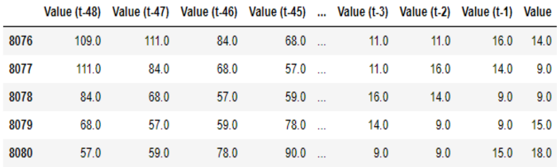
IV. METHODS

The focus was to train the LSTMs and then combine them into a meta model for every sensor separately. Generally, we can divide our experiments in two separate approaches: Univariate and Multivariate (describing the type of the LSTMs used in the first stage before the Meta models).

A. *Univariate*

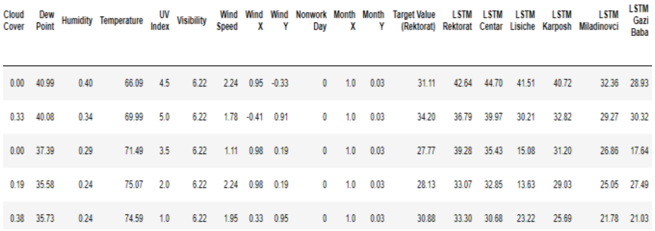
In this approach, we used only the sensor values from TRAIN1, without the *extras* to train five univariate LSTMs on all sensors separately, with a time lag of 48 hours, while optimizing the hyperparameters (number of units, hidden layers, activation functions, regularization, optimizers and loss functions) with the help of a validation set extracted from TRAIN1 as a hold-out. Next, we used the models to make predictions for the TRAIN2. We then combined these predictions from the LSTMs with the *extras* dataset acquiring a train set for the Meta model. This train set contained 18 features (12 from the extras and the 6 separate LSTM predictions for all sensors) and one target for every Meta model. For instance, the Meta model for Centar would contain LSTM predictions for Centar, Lisiche, Miladinovci, Karposh, Gazi Baba and Rektorat (along with the *extras*), but the target value would be the actual Centar measurement on the exact timestamp from the sensor. From Fig. 4 and Fig. 5, we see the format of the univariate LSTM train sets and the Meta train sets respectively. Fig. 6. shows the complete architecture of the first approach

All meta models are Support Vector Regressors with a gaussian kernel [10], trained using the Scikit-Learn machine learning library [8] and optimized on their three hyperparameters (C, epsilon and gamma) using 10-fold cross validation.



	Value (t-48)	Value (t-47)	Value (t-46)	Value (t-45)	...	Value (t-3)	Value (t-2)	Value (t-1)	Value
8076	109.0	111.0	84.0	68.0	...	11.0	11.0	16.0	14.0
8077	111.0	84.0	68.0	57.0	...	11.0	16.0	14.0	9.0
8078	84.0	68.0	57.0	59.0	...	16.0	14.0	9.0	9.0
8079	68.0	57.0	59.0	78.0	...	14.0	9.0	9.0	15.0
8080	57.0	59.0	78.0	90.0	...	9.0	9.0	15.0	18.0

Fig. 4. Train data for Univariate LSTM (lag 48)



Cloud Cover	Dew Point	Humidity	Temperature	UV Index	Visibility	Wind Speed	Wind X	Wind Y	Nonwork Day	Month X	Month Y	Target Value (Rektorat)	LSTM Rektorat	LSTM Centar	LSTM Lisiche	LSTM Karposh	LSTM Miladinovci	LSTM Gazi Baba
0.00	40.99	0.40	66.09	4.5	6.22	2.24	0.95	-0.33	0	1.0	0.03	31.11	42.64	44.70	41.51	40.72	32.36	28.93
0.33	40.08	0.34	69.99	5.0	6.22	1.78	-0.41	0.91	0	1.0	0.03	34.20	36.79	39.97	30.21	32.82	29.27	30.32
0.00	37.39	0.29	71.49	3.5	6.22	1.11	0.98	0.19	0	1.0	0.03	27.77	39.28	35.43	15.08	31.20	26.86	17.64
0.19	35.58	0.24	75.07	2.0	6.22	2.24	0.98	0.19	0	1.0	0.03	28.13	33.07	32.85	13.63	29.03	25.05	27.49
0.38	35.73	0.24	74.59	1.0	6.22	1.95	0.33	0.95	0	1.0	0.03	30.88	33.30	30.68	23.22	25.69	21.78	21.03

Fig. 5. Train data for meta model with Univariate LSTM (Rektorat)

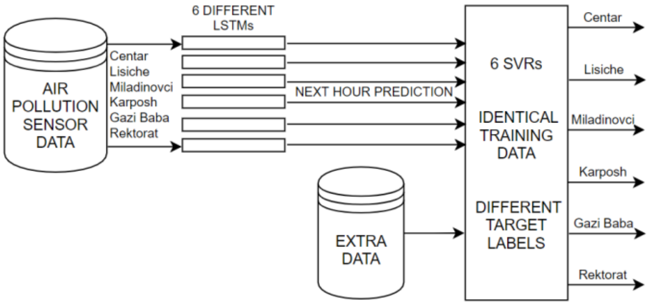


Fig. 6. Univariate architecture

B. *Multivariate*

The idea of the second approach was to use the *extras* from the beginning, by utilizing the multivariate system of the LSTMs implemented in Keras [9]. We used TRAIN1 with the 12 *extras* features along with the sensor values with a time lag of 3, granting a total of 39 features for every input. Since, Keras uses matrix 3D representation for multivariate problems, our vectors had the shape *of (#train_examples, 3, 13).* Having this format, we trained 6 LSTM multivariate models for every sensor and then we combined these into meta models for each respective sensor (Fig. 7). From Fig. 8 and Fig. 9 we see the format of the multivariate LSTM train sets and their respective meta train sets. Even though the two main approaches were quite the opposite, the results from the two separate methods were not significantly different.

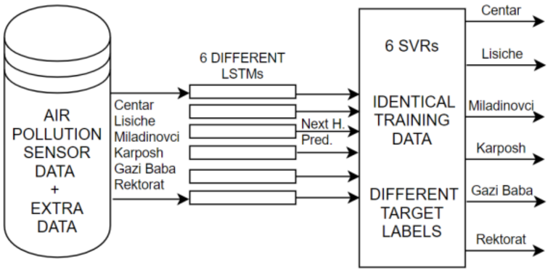
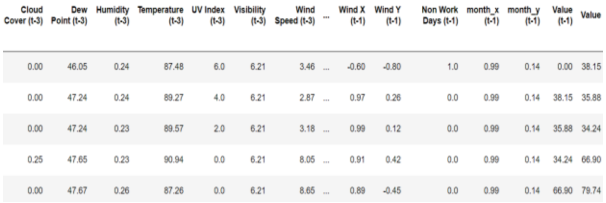
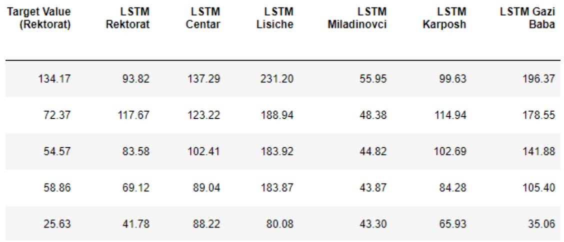


Fig. 7 Multivariate architecture



Cloud Cover (t-3)	Dew Point (t-3)	Humidity (t-3)	Temperature (t-3)	UV Index (t-3)	Visibility (t-3)	Wind Speed (t-3)	...	Wind X (t-1)	Wind Y (t-1)	Non Work Days (t-1)	month_x (t-1)	month_y (t-1)	Value (t-1)	Value
0.00	46.05	0.24	87.48	6.0	6.21	3.46	...	-0.60	-0.80	1.0	0.99	0.14	0.00	38.15
0.00	47.24	0.24	89.27	4.0	6.21	2.87	...	0.97	0.26	0.0	0.99	0.14	38.15	35.88
0.00	47.24	0.23	89.57	2.0	6.21	3.18	...	0.99	0.12	0.0	0.99	0.14	35.88	34.24
0.25	47.65	0.23	90.94	0.0	6.21	8.05	...	0.91	0.42	0.0	0.99	0.14	34.24	66.90
0.00	47.67	0.26	87.26	0.0	6.21	8.65	...	0.89	-0.45	0.0	0.99	0.14	66.90	79.74

Fig. 8. Train data for Multivariate LSTM (lag 3)



Target Value (Rektorat)	LSTM Rektorat	LSTM Centar	LSTM Lisiche	LSTM Miladinovci	LSTM Karposh	LSTM Gazi Baba
134.17	93.82	137.29	231.20	55.95	99.63	196.37
72.37	117.67	123.22	188.94	48.38	114.94	178.55
54.57	83.58	102.41	183.92	44.82	102.69	141.88
58.86	69.12	89.04	183.87	43.87	84.28	105.40
25.63	41.78	88.22	80.08	43.30	65.93	35.06

Fig. 9. Train data for meta model with Multivariate LSTM (Rektorat)

V. Results and Future Work

A. *Results*

RMSE (Root Mean Square Error) results are shown in Table I for every sensor and its respective four main predictions for the next hour: Univariate LSTM, Multivariate LSTM, meta derived from Univariate LSTMs and meta derived from Multivariate LSTMs. We can see that the meta models give a huge boost to both univariate and multivariate approaches for all sensors, while not showing a significant difference between both methods. Since the sensor values vary from 0 to 1000, being able to predict with RMSE around 10 could be very useful. Fig. 10 shows hourly timestep in subset of the predicted year (2018) from the test set for Centar's sensor. It is important to notice that predicting more than one hour ahead is possible in both cases. Univariate multistep prediction would utilize a circular movement going forward with the previous 47 hours plus the new predicted value from the meta, combining them with a forecasted weather data from an API like Darksky. The multivariate approach also allows the circular movement predictions but could easily implement immediate prediction for more hours since the LSTMs use the weather data instead of the meta, having no need for circular patterns forward in time.

TABLE I. PREDICTION RESULTS FOR THE NEXT HOUR

Sensor	**Root Mean Square Error (RMSE)**			
	Univariate LSTM	***Multivariate LSTM***	***Univariate Meta***	***Multivariate Meta***
Centar	21.784	21.028	9.872	12.749
Lisiche	37.606	37.957	30.452	31.242
Miladinovci	23.728	24.558	20.528	21.697
Karposh	22.714	24.232	11.34	14.375
Gazi Baba	38.922	39.953	29.213	31.370
Rektorat	32.417	35.063	26.231	28.868

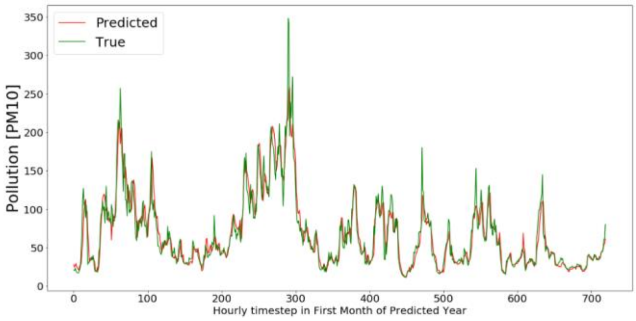


Fig. 10. Meta model Prediction vs. Actual

B. *Future Work*

Finding the optimal way to predict at least 24 hours straight is possible and is crucial for many stakeholders (government, ecologists, non-profit organizations etc.) for managing resources, detecting polluters, measuring successfulness of the safety measures and many more, but also it is important information for the everyday life of the citizens of North Macedonia. There are many experiments that can be done in the future on managing the LSTMs, but also in combining the Meta models in many different manners. If we would use multi-target LSTMs (predicting 24 hours in the future instead of one), we would have multiple outputs of every LSTM model, thus the attribute combinations for the Meta models are huge.

We could experiment in that time interval of 24 hours and find patterns that would be very useful for a precise prediction. For example, we could use future predictions of the LSTMs (10 hours ahead) as an input to the Meta model for just one hour ahead or vice versa.

There is also a room for improvement on the data gathering and quality, since there are many weather attributes that were not available for us for the time being (for example: cumulative rainfall and cumulative snowfall for the past hour, day, week etc.). We still have some ideas for data preprocessing and transforming the date and time parameters into some more useful, but also discovering some new ones too.

Even though the models recognize the overall pattern, the amount of data is a huge factor for the LSTMs, so making a system that can train daily on the new everyday data is our top priority for this project.

VI. Conclusion

Predicting air pollution accurately is possible, especially by using the right weather and air pollution sensor data from robust sensors placed on noiseless surroundings. In this paper, we presented a method for utilizing this correlation between sensors on different location by combining predictions from different models into one with the usage of Meta models. By training these correctors of the LSTMs, we drastically increased the prediction capabilities, thus emphasizing the connection between the measurements of different sensors, gaining RMSE results below 20 for most of the Meta models.

Because long-term prediction tasks are naturally more difficult, they require more relevant historical data, including optimum time lags, which adds another layer of optimization of the LSTM model. This means that many hyperparameters, such as batch size and number of LSTM cells, may still be optimized to return a lower RMSE for longer future forecasting. The hope is that by leveraging as much time series data as possible we can create stronger weights in the RNN, based on the sequence dependencies. As mentioned above, longer prediction times can help cities in policymaking and resource allocation, but more importantly, can help in the main battle against air pollution in order to solve this problem for humanity and all living species on the planet once and for all.

References

[1] Bobak, M & Leon, David. (1992). Air pollution and infant mortality in the Czech Republic, 1986-88. Lancet. 340. 1010-4. 10.1016/0140-6736(92)93017-H.

[2] Dockery, Douglas & Pope, C & Xu, Xuebing & Spengler, Jack & H. Ware, James & E. Fay, Martha & G. Ferris, Benjamin & Speizer, Frank. (1994). An Association Between Air Pollution and Mortality in Six U.S. Cities. The New England journal of medicine. 329. 1753-9. 10.1056/NEJM199312093292401.

[3] World Health Organization. (2018). Air pollution and child health: prescribing clean air: summary. World Health Organization. http://www.who.int/iris/handle/10665/275545. License: CC BY-NC_SA 3.0 IGO.

[4] Dimovska, Mirjana. (2018). Assessing Health Impact of Air Pollution in Macedonian Cities. Biomedical Journal of Scientific & Technical Research. 10. 10.26717/BJSTR.2018.10.001887.

[5] Bengio, Y & Simard, Patrice & Frasconi, Paolo. (1994). Learning long-term dependencies with gradient descent is difficult. IEEE transactions on neural networks / a publication of the IEEE Neural Networks Council. 5. 157-66. 10.1109/72.279181.

[6] Hochreiter, Sepp & Schmidhuber, Jürgen. (1997). Long Short-term Memory. Neural computation.9.1735-80.10.1162/neco.1997.9.8.1735.

[7] Wes McKinney. (2010). Data Structures for Statistical Computing in Python, Proceedings of the 9th Python in Science Conference, 51-56

[8] Pedregosa. (2011). Scikit-learn: Machine Learning in Python et al., JMLR 12, pp. 2825-2830,

[9] P.W.D. Charles, Project Title, (2013), GitHub repository, https://github.com/charlespwd/project-title

[10] Basak, Debasish & Pal, Srimanta & Chandra Patranabis, Dipak. (2007). Support Vector Regression. Neural Information Processing – Letters and Reviews. 11.

Deploying production-grade Kubernetes cluster

Vojdan Kjorveziroski*, Panche Ribarski†
Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje
Email: *vojdan.kjorveziroski@finki.ukim.mk, †panche.ribarski@finki.ukim.mk

***Abstract*—Deploying a production-grade Kubernetes cluster is a challenging feat, mainly because of all the different services that need to be integrated with each other. We examined Kubespray, which is an open-source project whose purpose is to automate the deployment of stable clusters, as well as ease the future administration and lifecycle management along the way. Additional components can be installed during the deployment process that enrich and improve the cluster capabilities. We used Kubespray and together with other open-sources projects we created production-grade Kubernetes cluster ready for real deployment scenarios.**

***Index Terms*—Kubernetes, K8S, Docker, Kubespray, Helm, ELK, monitoring, automation, containers, orchestration**

I. Introduction

Nowadays there is an increasing trend in the computer engineering world to adopt the cloud as a native place for the execution of many workloads, ranging from high performance computations to hosting lightweight demo student projects made as part of some course assignment. There are many reasons that support this notion of using someone else's infrastructure for your work. One of them is the increased reliability, since the end user is no longer in charge of managing the hardware and affordability, because the primary method of billing is pay-as-you-go, meaning that users are obliged to pay only for the resources that they have used as many cloud providers even have the option of billing per minute or even per second for certain services. Other reason, scalability, in terms of available resources is no longer a problem, additional storage or memory is just a few clicks away. However, many companies and even individuals are either not satisfied with the current offerings from the major cloud providers, have industry or policy bound restrictions that prevent them from using public clouds, think that they can implement a more cost-effective solution by using their own hardware or simply want to have more control over the whole process and learn something new along the way. In all these cases, the usual options of implementing the infrastructure for hosting multiple applications or services is to either deploy individual virtual machines and assign them to different teams or departments or go with the container route, where each application will reside in its own dedicated environment. No matter what option is chosen, some abstraction layer in terms of software will have to be deployed that will need to manage the underlying hardware and allow for easy and convenient provisioning and deprovisioning of either virtual machines or containers. These days containers are the preferred choice for many workloads, mainly because they are lightweight and easily disposable, yet at the same time provide sufficient isolation. Manual deployment of containers quickly becomes a burdensome process, so a container orchestration system is almost always needed. Kubernetes[1][2] has distinguished itself as a leader in this space, mainly because of its extensibility, large community and large feature set. However, all of these benefits result in a complex installation process, large amounts of prerequisites and dependencies, and elaborate lifecycle management. The common solution to the aforementioned problems is to have a well-tested automated solution which can offer a reproducible and stable deployment each time it is invoked. There are multiple such solutions when it comes to the installation of Kubernetes, one being Kubespray[3], an open-source project that is centered around Ansible[4][5] playbooks and roles that provisions production grade clusters with minimal end-user interaction. The purpose of this paper is to explain the benefits of using Kubespray while exploring the customizability that it offers and options for the future lifecycle management of the deployed clusters. First, a general overview of the Kubernetes platform is given, and then the Kubespray project is explained in details, along with possible additions to the cluster that improve its functionality, either deployed by Kubespray itself during the initial setup, or manually at a later point in time.

II. Automated deployment of production grade clusters

Kubernetes is a container orchestration platform that can operate on top of many container runtimes, most popular of which are Docker[6][7] and Rkt. A deployment of an application is done using multiple manifests that represent YAML based text files, where using a predetermined syntax and options the desired environment for the application is described. These manifests are then submitted to the Kubernetes API server and are eventually translated to low level actions such as bringing up a container or issuing a storage request to a remote system. The API server is extensible, so cluster administrators can define custom resource types that include their own logic and interact with systems that are not natively supported.

Since the deployment of a Kubernetes cluster involves many dependencies and is a complicated feat, multiple open source projects exist whose purpose is to make the process easier. One such example is Kubespray, a solution based on Ansible playbooks and roles that automate every aspect of the Kubernetes cluster deployment. The end-user is only required to install the dependencies that include a couple of Python libraries and Ansible itself in order to deploy a cluster.

A. Cluster requirements

Kubernetes clusters are comprised of two types of nodes, masters and workers. Master nodes are hosting the Kubernetes API server and are responsible for managing the cluster. In order to achieve high availability, multiple master nodes can be deployed and an uninterrupted cluster operation will be ensured as long as a majority of monitors can be formed. Workers can be added or removed from the cluster at any point in time, depending on the resource requirements. The worker nodes are directly controlled by the master nodes. As a persistent backend the etcd key-value store is used and all the cluster information including the deployed manifests are stored there. Similar to the Kubernetes API server, etcd can also operate in a high-avialability mode, if more than half of the nodes are available. Since etcd also has official Docker images available, the cluster operator can choose to either run it in a standalone fashion, installed as any other software package or inside a Docker container.

Publishing of web applications is done using an Ingress controller[8] which effectively acts as a reverse proxy to any container that exposes an HTTP port. The configuration of the ingress controller, such as what endpoints to use, whether SSL is enabled, what redirects to use; is done using annotations, simple key value pairs in the YAML file that describes the Ingress resource. Since Kubernetes is such an extensible platform, there are multiple Ingress controllers to choose from, for example NGINX[9] or Traefik[10]. The NGINX ingress controller is natively supported by Kubespray. If an application that does not use the HTTP protocol needs to be exposed to the outside world, then there are multiple options, some of which are binding it to a specific worker node and accessing it through it exclusively or, a more attractive and versatile option is to deploy a load balancer[11] addon which can automatically assign either public or private IP addresses from a given subnet. One such addon is MetalLB[12], which from recently is also supported by Kubespray and can automatically be provisioned. Whenever a LoadBalancer resource is defined within the Kubernetes API, a private IP address from a predefined range is assigned to the service. Furthermode, BGP is also supported by MetalLB, in environments where assignment of public IP addresses is needed.

Networking is a major topic when it comes to Kubernetes cluster deployments, since there are so many options and extensions available. An important thing to note is that the chosen network plugin is responsible for the isolation of the different containers running in the cluster. Kubernetes natively supports a NetworkPolicy[13] resource type that can be specified like all other resources, using a YAML file. With this, granular network rules for an application can be specified, such as what ports are opened and from what other containers or namespaces they can be accessed. More details regarding networking plugins and deployment options are given in the Additional tools section.

B. Initial setup

Once the project has been downloaded from the source code repository, an inventory file needs to be created in which the IP addresses of all the nodes that will take part in the clusters will be entered. Kubespray exploits the concept of groups that Ansible provides, where multiple hosts can be put together in a group so that they can be referenced using a common name, in order to designate the various roles that the nodes will have inside the cluster, such as whether they will be master nodes, worker nodes, etcd nodes, or have some other role. Once the inventory has been defined, the cluster deployment can be customized by editing the default values of the various predefined parameters. Using simple switches and statements, the Kubernetes version can be specified, along with the desired networking plugin, container runtime, ingress controller, and storage addons. Finally, with the execution of the Ansible playbook, Kubespray will perform any prerequisite checks on the nodes and install all of the missing requirements before deploying the cluster. After the script has finished with its execution, the administrator should be able to login into the master nodes and have a fully functional Kubernetes cluster.

C. Lifecycle management

Kubespray would not be so enticing if it did not support some sort of lifecycle management, ensuring that the cluster functions properly after it has been deployed. The following non-distrubtive operations are possible with Kubespray: addition and removal of worker, master and route reflector nodes, API server certificate renewal and rotation, cluster upgrade and cluster tear-down. The most beneficial of these is the automatic cluster upgrade, where not only is the Kubernetes version upgraded, but also all of the prerequisites and addons, such as the etcd store, the container runtime and the networking plugins. Furthermore, the addition and removal of worker nodes is simple, fast, and it does not require user intervention, meaning that it can be automated with some monitoring system, where new nodes would be deployed once the cluster load reaches a certain level.

III. Additional tools

In the Cluster Requirements section, we briefly mentioned some of the Kubernetes addons that are supported by Kubespray. In the following paragraphs we provide a more thorough explanation of some of them.

A. Kubernetes package management

The definition of several YAML manifest files for a deployment of a single service can quickly become tiresome, since most of the time the required changes are very small in comparison to the complete files. Helm[14] is a package manager for Kubernetes that tries to solve this problem. It uses the concept of charts, where a chart represents a service that needs to be deployed, such as a content management system (CMS) application. Using a templating language, a set of default parameters are inserted in the manifest files and the user can choose to override only the desired values.

Helm requires a dedicated component, Tiller, to be installed in the cluster itself, and Kubespray fully supports its complete deployment.

B. Storage

There are many options for integrating the Kubernetes cluster with standalone storage systems. Kubernetes has a resource called a persistent volume claim[15], where such a claim can be created by an application and the default storage provisioner that is deployed in the cluster is required to fulfill the request. The way in which the request will be serviced is up to the plugin, whether a new block device needs to be set up, a shared file system mounted or a new bucket on some object storage created. For test clusters an NFS provisioner[16] might be sufficient, where all the volume claims will be serviced by mounting a specific directory from a given NFS server. However, this is not scalable since the performance of NFS is not optimal for such deployments. A popular option is to integrate the Kubernetes cluster with Ceph[17][18], a storage system that offers both block and object storage and a shared file system. Since Ceph is also a distributed application that pools disks present on different hosts, the end-user needs to decide how the Ceph cluster will be managed. One option is to have a separate Ceph cluster, preferably on bare metal machines, and integrate it with the Kubernetes cluster with volume provisioning plugins. These plugins watch for any new volume claims and execute the necessary commands on the Ceph cluster in order to deploy a new block device or mount the shared file system. Another option is to use Rook[19], a Kubernetes native storage solution that deploys a Ceph cluster inside Kubernetes. The initial deployment and future management of the storage cluster is done through YAML files that are then submitted to the Kubernetes API server. One of the main benefits of Rook is that the operation of the storage system is completely abstracted from the end-user, so no knowledge of Ceph is required. However, this can also be seen as a drawback; Kubernetes upgrades now become much more complicated, since bringing down Kubernetes nodes might bring the whole storage cluster down, so careful planning and testing is needed. On the other hand, while administering a standalone Ceph cluster requires much more resources both in terms of people and hardware, it is a more robust solution and provides a nice separation of concerns, where multiple complex systems are not intertwined with each other.

C. Networking

There are two main types of networking plugins available for Kubernetes and they are layer 2 or layer 3. An example of a layer 2 plugin is Flannel[20], which creates a single flat network shared by all future containers. Another option is to use separate networks and with the help of a routing algorithm bridge these together. A plugin that works in this way is Calico[21] and it uses BGP for routing between the different subnets in the overlay network. The benefits of using Flannel or any other L2 plugin is that it is easier to configure and simpler to administer, but it is not as scalable as using an L3 solution. On the other hand, Calico supports the concept of BGP route reflectors, which can be used in very large clusters. In this mode, instead of forming a BGP full mesh topology, a separate node outside the cluster is configured to run as a standalone BGP route reflector and all the cluster nodes peer only with it. For redundancy, multiple route reflectors can be deployed. Both of these plugins as well as a few others are supported by Kubepsray and the user has an option of manually choosing between them.

D. Monitoring and logging

Monitoring of the Kubenetes cluster and the individual services that are deployed as containers within it is an important task and there are many different solutions that can be implemented. One of the most popular options regarding performance metrics is to use Prometheus[22], a dedicated software which crawls different endpoints that expose system metrics. These metrics can then either be persisted in a separate time series database or cached by Prometheus itself. The visualization of the gathered values can be done using a separate web application application, one example being the widely popular Grafana[23]. When it comes to logging, the usual choice not only when working with Kubernetes, but also in other areas is to use the Elasticsearch, Logstash and Kibana (ELK)[24] stack. An important thing to note is that cluster administrators should not only be interested in the logs from the cluster nodes themselves, but also from the containers that are run within the cluster. In order to achieve this, there are multiple Kubernetes addons that are able to send the log output of the containers either directly to Elasticsearch or to Logstash for further processing. One such example is Fluentd[25]. Once the logs have been persisted, graphical visualizations and dashboards can be created using Kibana, which is a web application that connects to Elasticsearch and provides browsing and visualization options for the gathered logs.

IV. Deployment of production-grade cluster with Kubespray on the Nebula platform

We tested Kubespray in an attempt to bring up a production-grade Kubernetes cluster on virtual machines residing on the OpenNebula cloud computing platform. In this section we describe the deployment process, along with the choices that we have made and additional applications that were installed in order to ensure the reliability and availability of the cluster.

A. Hardware

We have chosen a rather minimal initial setup that will offer the option for non-disruptive upgrades to the cluster and tolerate a single node failure, with the possibility to add new nodes as needed. The storage has been completely decoupled from Kubernetes and an external Ceph cluster is utilized. This setup allows both the computational power to be increased by adding new worker nodes, or to improve the resiliency by deploying additional masters, without the need for any major architectural changes. The separation of storage from the

Purpose	CPU	Memory	Storage
Kubernetes Master #1	4	4GB	40GB
Kubernetes Master #2	4	4GB	40GB
Kubernetes Master #3	4	4GB	40GB
Kubernetes Worker #1	8	8GB	60GB
Kubernetes Worker #2	8	8GB	60GB
Ceph #1	2	4GB	40GB
Ceph #2	2	4GB	40GB
Ceph #3	2	4GB	40GB
Ceph #4	4	4GB	40GB + 1TB
Ceph #5	4	4GB	40GB + 1TB
Ceph #6	4	4GB	40GB + 1TB

cluster allows easy maintenance of both the cluster itself and the underlying virtual machines. Additionally, a catastrophic failure in the Kubernetes cluster will not affect the data, all that will be needed to restore normal operation is to recreate the necessary Kubernetes resources and remap the persistent volumes to the RBD volumes on Ceph. If Rook had been used, then this would have been much more complicated, and a Kubernetes failure would have probably meant corrupton of the data as well.

The specification for the virtual machines are:

B. Storage cluster deployment

The storage cluster has been deployed with the officially supported Ansible playbooks from the Ceph community. The process is very similar to Kubespray, where the cluster ”skeleton”, the roles for each virtual machine, are given in the inventory files and any further tweaks regarding the operation of the cluster is done by modifying the default Ansible variables. We have provisioned six Ceph nodes, where the first three (1-3) serve as both monitor and manager nodes in order to ensure the high-availability of the cluster and the others as object storage device (OSD) hosts, where, currently, each of them hosts a single OSD, with the option of adding additional ones at a later point in time. Additional monitors or managers can also be added whenever needed.

C. Kubernetes cluster deployment

For the initial deployment, five Kubernetes nodes were deployed, where three of them are masters that also function as etcd servers and the rest are worker nodes. No route reflector was used, since the initial number of nodes is low, but it remains an option for future expansion. At least two additional nodes will be required in order to ensure the high-availability of the route reflector. The Kubernetes addons that were installed by Kubespray are: NGINX ingress as the cluster ingress controller, CoreDNS as the in-cluster DNS server, Calico as the networking plugin, Helm and Tiller with TLS support, Kubernetes dashboard for web access to the cluster resources. The communication between the nodes and the Ceph cluster is done through a private VLAN dedicated to the project. Official support for the MetalLB load balancer was added to Kubespray after the cluster was deployed, so manual installation was needed. Regarding the RADOS block device (RBD) provisioner that allows the mapping of Kubernetes volumes to RBD images, there is an active pull request[26] that is yet to be merged into the main branch of Kubespray at the time of this writing. Once it is approved and merged, that process could be automated as well.

D. Additional tools and addons

In order to be production ready, we needed a way to monitor the cluster resources, receive alerts for any anomalies, and centralize the logging for all the different services that are installed. To achieve this, we deployed Prometheus-Operator in order to monitor the cluster and receive alerts, and the EFK stack (Elasticsearch, Fluentd, Kibana) to centralize the logs.

Prometheus operator[27] is a project that offers Kubernetes native monitoring. Once deployed, it installs Prometheus, Alertmanager[28], and Grafana inside the cluster. Prometheus is responsible for scraping the metrics of the various Kubernetes components, such as the API server, the scheduler and the controller, as well as any additional services that are deployed inside the cluster and expose Prometheus metrics. The metrics are by default persisted inside Prometheus, with the option of using an external database as well. Alertmanager comes preconfigured with some default alerts that are sent when cluster resources are low, or when critical Kubernetes components are unavailable. Grafana comes preloaded with few default dashboards, where various graphs are available that showcase the current cluster state. The cluster administrator can get detailed information on all possible levels, from individual pods, to cluster nodes, to the cluster itself. One benefit of Prometheus operator over an independent deployment is that it installs custom resource definitions[29] (CRDs) inside the cluster, and these can be used for defining YAML files that describe what additional services need to be monitored, the same way that native Kubernetes resources are defined, for example Deployments.

Log management is made possible with the EFK stack, where Fluentd is deployed as a DaemonSet on all of the available Kubernetes nodes. Fluentd tails all of the container logs as well as the host logs, preprocess them, enriches them with Kubernetes specific data, such as the namespace to which the given container belongs, or the node where it runs and then sends these to Elasticsearch for long term storage and indexing. Kibana is a web application that connects to Elasticsearch and can monitor the logs in real-time and even visualize them using different graph types. Since no retention policy can be configured in Elasticsearch, an additional tool needs to be used, in this case Elasticsearch Curator[30] which has the option of deleting indices on predefined conditions, one being time. Each day, a new index is created in Elasticsearch that will contain the logs for that day. Deleting the index removes the logs for that period and frees up the used storage space. Elasticseach Curator works as a Kubernetes Cronjob that can be executed at a desired interval.

V. Conclusion

Kubespray is already a major player in the Kubernetes world and is even one of the officially recommended ways to deploy Kubernetes. The large user base, along with a rich feature set, active development and support for lifecycle management make it one of the best choices for new deployments. Not all of the Kubernetes extensions mentioned in this paper are natively supported by Kubespray, but the easy customizability of the Ansible playbooks and roles means that end users can extend them with minimal effort.

References

[1] M. Lukša, *Kubernetes in Action*. Manning Publications Company, 2018.

[2] "Kubernetes - production-grade container orchestration," https://kubernetes.io/docs/home/, accessed: 2019-04-13.

[3] "Kubespray - deploy a production ready kubernetes cluster," https://kubespray.io/, accessed: 2019-04-13.

[4] L. Hochstein and R. Moser, *Ansible: Up and Running: Automating Configuration Management and Deployment the Easy Way*. " O'Reilly Media, Inc.", 2017.

[5] "Ansible documentation," https://docs.ansible.com/, accessed: 2019-04-13.

[6] J. Turnbull, *The Docker Book: Containerization is the new virtualization*. James Turnbull, 2014.

[7] "Docker provides a way to run applications securely isolated in a container, packaged with all its dependencies and libraries," https://docs.docker.com/, accessed: 2019-04-13.

[8] "Kubernetes ingress - kubernetes api object that manages external access to the services in a cluster," https://kubernetes.io/docs/concepts/services-networking/ingress/, accessed: 2019-04-13.

[9] "Nginx ingress controller for kubernetes," https://github.com/kubernetes/ingress-nginx, accessed: 2019-04-13.

[10] "Traefik - kubernetes ingress controller," https://docs.traefik.io/user-guide/kubernetes/, accessed: 2019-04-13.

[11] "Kubernetes load balancer - kubernetes api object that integrates with external load balancers," https://kubernetes.io/docs/concepts/services-networking/#loadbalancer, accessed: 2019-04-13.

[12] "Load balancer implementation for bare metal kubernetes clusters, using standard routing protocols," https://metallb.universe.tf/, accessed: 2019-04-13.

[13] "Networkpolicy - kubernetes resource which specifies how groups of pods are allowed to communicate with each other and other network endpoints," https://kubernetes.io/docs/concepts/services-networking/network-policies/, accessed: 2019-04-13.

[14] "Helm - the package maanger for kubernetes," https://helm.sh/, accessed: 2019-04-13.

[15] "Kubernetes resource which acts as an abstraction to other storage systems," https://kubernetes.io/docs/concepts/storage/persistent-volumes/#persistentvolumeclaims, accessed: 2019-04-13.

[16] "Kubernetes nfs dynamic provisioner," https://github.com/kubernetes-incubator/external-storage/tree/master/nfs, accessed: 2019-04-13.

[17] N. Fisk, *Mastering Ceph*. Packt Publishing Ltd, 2017.

[18] "Ceph uniquely delivers object, block, and file storage in one unified system," http://docs.ceph.com/docs/master/, accessed: 2019-04-13.

[19] "Rook - storage orchestration for kubernetes," https://rook.io/, accessed: 2019-04-13.

[20] "Flannel - a network fabric for containers, designed for kubernetes," https://github.com/coreos/flannel, accessed: 2019-04-13.

[21] "Calico is an open source networking and network security solution for containers, virtual machines, and native host-based workloads," https://docs.projectcalico.org/v3.6/introduction/, accessed: 2019-04-13.

[22] J. Turnbull, *Monitoring with Prometheus*. Turnbull Press, 2018.

[23] "Grafana is an open platform for beautiful analytics and monitoring," https://grafana.com/, accessed: 2019-04-13.

[24] "Elasticsearch, logstash, kibana, open-source logging stack," https://www.elastic.co/elk-stack, accessed: 2019-04-13.

[25] "Fluentd - open source data collector," https://www.fluentd.org/, accessed: 2019-04-13.

[26] "Add rbd provisioner addon support to kubespray," https://github.com/kubernetes-sigs/kubespray/pull/3668#issuecomment-480623695, accessed: 2019-04-13.

[27] "The prometheus operator creates, configures, and manages prometheus monitoring instances." https://coreos.com/operators/prometheus/docs/latest/, accessed: 2019-04-13.

[28] "Alertmanager handles alerts sent by client applications such as the prometheus server," https://prometheus.io/docs/alerting/alertmanager/, accessed: 2019-04-13.

[29] "Custrom resource definitions - kubernetes documentation," https://kubernetes.io/docs/concepts/extend-kubernetes/api-extension/custom-resources/#customresourcedefinitions, accessed: 2019-04-13.

[30] "Elasticsearch curator eases the management of elasticsearch indices," https://www.elastic.co/guide/en/elasticsearch/client/curator/current/about.html, accessed: 2019-04-13.

Exploratory study into melanoma skin cancer detection

Andrej Jovanov, Ivana Ristova, Sonja Gievska
Faculty of Computer Science and Engineering
Ss.Cyril and Methodius University, Skopje, Macedonia
Email: andrej.jovanov@students.finki.ukim.mk, ristova.ivana.1@students.finki.ukim.mk, sonja.gievska@finki.ukim.mk

***Abstract*—Melanoma is one of the most aggressive and lethal tumors accounting for about 80% of deaths from skin cancer. The analysis of dermoscopic images has proven useful in the process of confirmation of presumptive diagnoses. Our research objective was to investigate a number of methods and algorithms suitable for image segmentation and identification of melanoma varieties in clinical dermoscopic images. The performance of the proposed method for image segmentation has been evaluated on the test dataset provided at the ISIC 2018: Skin Lesion Analysis Towards Melanoma Detection grand challenge, obtaining 0.491 Jaccard index score. A deep neural model that uses a pre-trained VGG16 model for melanoma detection, have achieved an average accuracy of 0.394 at the same challange, on the Task 3: Lesion Diagnosis test dataset.**

***Index Terms*—Image Processing, Segmentation, Classification, Neural Networks, Melanoma**

I. Introduction

Skin cancer is the most common type of cancer, non-melanoma accounts for over 5.4 million cases in more than 3.3 million people in the United States. In the last three decades it is estimated that the prevalence of malignant melanoma in Europe and the United States has almost tripled and more people have had skin cancer than all other cancers combined [1].

Melanoma is one of the most aggressive and lethal tumors with melanoma-induced mortality accounting for about 80% of deaths from skin cancer. Dermoscopy is most frequently used non-invasive and cost-effective way for detecting early-stage skin cancer. The analysis of dermoscopic images has proven to be useful in reducing the number of presumptive diagnoses that have to be histologically confirmed with skin biopsy [2].

Better image analysis can lead to early disease detection. Essential process for most medical image tasks is image segmentation. With simple and effective techniques, the region of interest can be extracted from the background which can lead to better classification and diagnosis [3].

In the last decade, a remarkable progress had been made in image recognition mainly due to the advances in the field of deep learning. Convolutional neural networks have been applied to a number of diagnostic problems in dermatology, from distinguishing between various types of malignant and benign skin diseases to segmentation of skin lesions. [4].

The main objective of this paper was to explore and evaluate the performances of simple segmentation techniques and convolutional neural networks for skin lesion classification (or melanoma detection).

II. Image Segmentation

A. Dataset for image segmentation

The dataset used for the experiments relating to image segmentation was included as ”Task 1: Lesion Boundary Segmentation”[1]at the ”ISIC 2018: Skin Lesion Analysis Towards Melanoma Detection“ [5][6]. Each of the 2594 images contains exactly one primary lesion, while other fiducial markers, smaller secondary lesions, or other pigmented regions should have been neglected. The distribution of images with benign and malignant skin lesions represent a modified “real world” setting i.e., there are more images with benign lesions than malignant lesions, with an over-representation of malignancies.

B. Artifacts removal

Empirical evidence has shown that preprocessing of images relating to removal of irrelevant artefacts and objects present in dermoscopy images could affect the performance on various tasks related to melanoma diagnosis [7]. Vignette frames, hair, ink-marks, scale-rulers and blood vessels are some of the obstacles that remain challenging for the tasks at hand, so we have opted for finding the most suitable solution for their detection and removal.

A large number of images in the ”Task 1: Lesion Boundary Segmentation” dataset contained dark circular frames caused by the strong illumination at the center of view because of the lighting from the dermoscope, or from the restricted diameter of the cannula on the dermoscope tip.

To remedy this problem, we have used the U-net approach using histogram equalization as preprocessing [8] that have been shown to work well for images with a vignette frame that is equally distant from the edges of an image. A modified version of the algorithm proposed in [8] was used. Instead of one, four circular masks, one for each corner of the image, were used. By having separate radius variable for each quadrant, we were able to apply different sizes of cropped circular masks on each corner of the image and successfully remove vignette frames regardless of the size, placement or shape of the vignette frame.

[1] https://challenge2018.isic-archive.com/task1/training/.

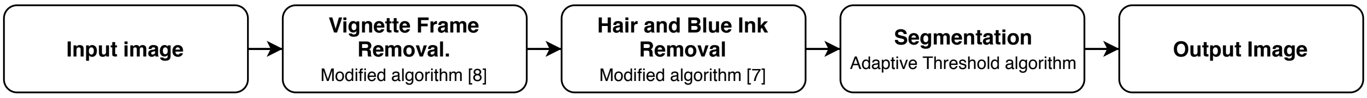


Figure 1. A diagram illustrating the steps of the image segmentation process.

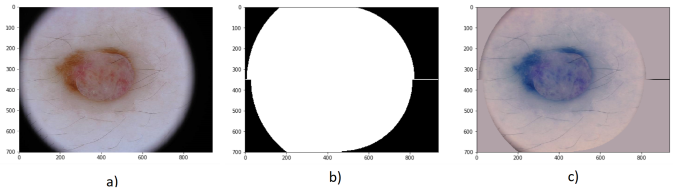


Figure 2. a) original image, b) vignette mask, c) result from vignette removal.

The process of removal of hair and scale-rulers from an image, started with masking of unwanted regions [7]. To detect the true hair edges, two thresholds were generated using the Adaptive Threshold algorithm for hysteresis thresholding stage in the Canny Edge detection method. The conventional thresholding techniques use a global threshold for all pixels, whereas adaptive thresholding changes the threshold value dynamically over the image. This was the key factor when choosing the segmentation technique. Subsequently, Canny Edge detection method is applied and the pixels lying between the two thresholds are classified as hair edges based on their connectivity, thus providing accurate results for images with unwanted artifacts. After multiple morphological dilations were applied, the extraction mask was created and the image inpaining algorithm developed by Alexandru Telea [9] was applied. Figure 3 shows the resulting images after the artifact removal process.

C. Image Segmentation

For image segmentation of preprocessed images, the Adaptive Thresholding was used, however instead of using one threshold value per image, the algorithm iterated over an image, calculating the threshold values for small regions. The largest contour area generated by the Adaptive Thresholding algorithm draws the final mask, as shown in Figure 4.

D. Image Segmentation Results and Discussion

The performance on the image segmentation challenge was measured as Jaccard index. For each image, a pixel-wise comparison of each predicted segmentation with the corresponding ground truth segmentation was used to calculate the Jaccard index. The final score for each image was computed as a threshold, i.e., if the Jaccard index was below 0.65 the score was 0. otherwise 1. The method we have proposed has achieved Jaccard Index score of 0.491 on the test dataset provided as ”Task 1: Lesion Boundary Segmentation” at the “ISIC 2018: Skin Lesion Analysis Towards Melanoma Detection” grand challenge. Comparison of the results with other research teams is not possible because of the difference in our goals. The objective of the task was automated predictions of lesion segmentation boundaries within dermoscopic images, while we have only used the dataset to evaluate our method for image segmentation.

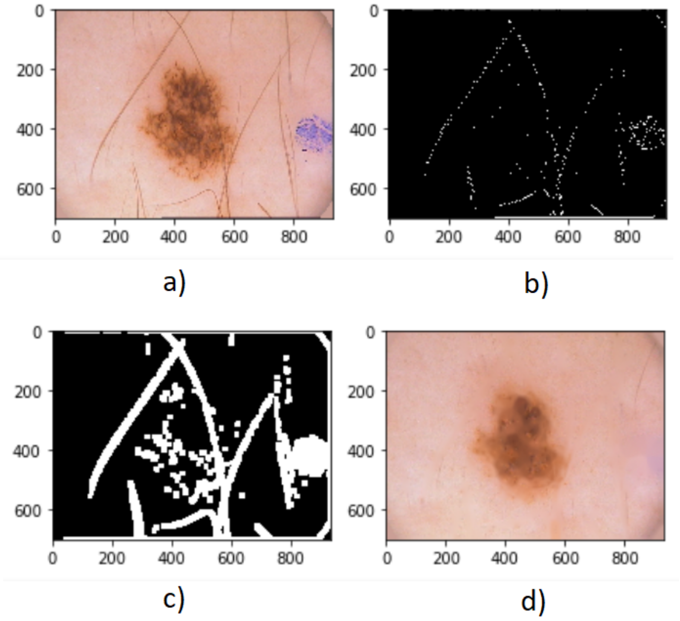


Figure 3. a) original image, b) Canny edge detection, c) Morphological transformations, d) result from hair/ink removal.

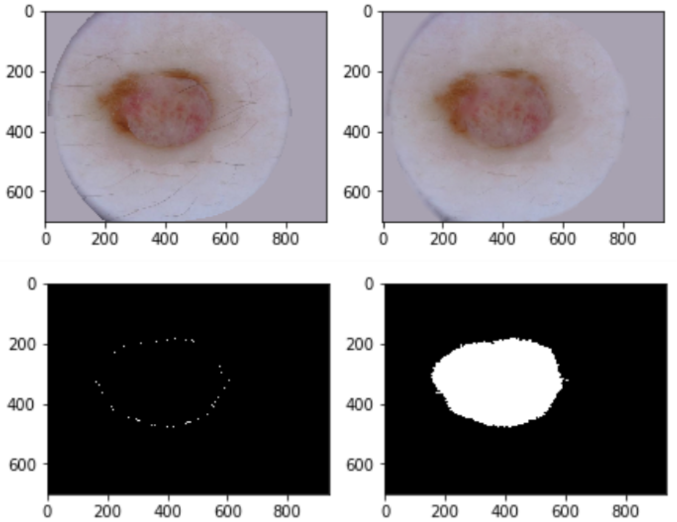


Figure 4. a) removed vignette frame, b) removed hair, c) best matching contour, d) segmentation result.

III. Image classification

A. Dataset for image classification

The dataset used for the experiments relating to image classification was included as ”Task 3: Lesion Diagnosis”[2]at the ”ISIC 2018: Skin Lesion Analysis Towards Melanoma Detection”[5][6]. The dataset consists of 10,015 images containing seven types (classes) of lesions, namely, melanoma (1113), melanocytic nevus (6705), basal cell carcinoma (514), actinic keratosis (327), benign keratosis (1099), dermatofibroma (115), and vascular lesions (142). The distribution of classes was not balanced.

B. Image classification using convolutional neural networks

For classification of seven types of lesions in images that are contained in the ”Task 3: Lesion Diagnosis” dataset, a well-known pre-trained convolutional neural network Visual Geometry Group 16 (VGG16) introduced by Visual Geometry Group from the University of Oxford [10] was used. VGG16 is a convolutional neural network model proposed by K. Simonyan and A. Zisserman [11], that have achieved 92.7% top-5 test accuracy on ImageNet dataset, which is a dataset of over 14 million images belonging to 1000 classes. The architecture of the 16-layer VGG16 model and the modified architecture used in our own experiments are shown in Figure 5 and Figure 6, respectively. Inspired by the VGG16-based model used for binary classification of lesions [12], we have used a modified version of the original pre-trained model adapted to serve as a 7-class classifier (one class for each type of lesion) with a [0,1] Softmax normalization function, interpreting the output scores as categorical probabilities. We were able to use the pre trained model and with shorter time of training to adjust the weights for our specific domain problem. For small dataset, it is advisable to use data augmentation techniques to increase the training dataset.

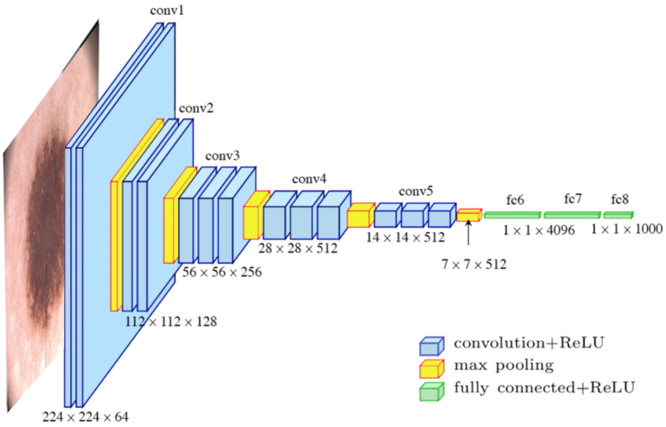


Figure 5. Standard VGG-16 network architecture as proposed in [10]

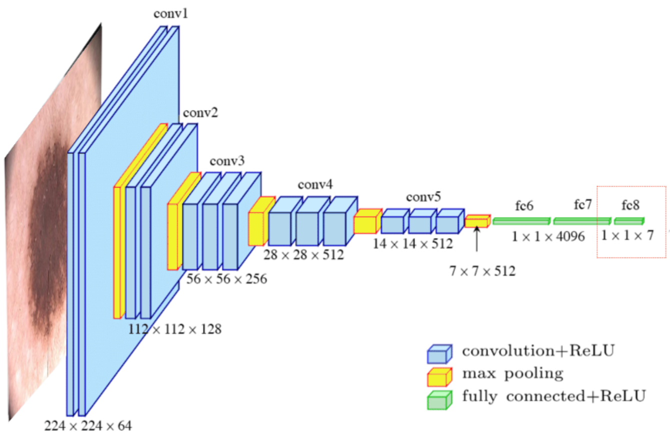


Figure 6. Modified VGG-16 network architecture as proposed in [12]

C. Data preparation and model validation technique

We have used a convolutional neural network (CNN) with 5-fold cross-validation [13], [14], [15] (which is a recommended approach when dealing with a limited quantity of data for training and testing. [15]). Cross-entropy loss was used to measure the error, while the learning rate was set to 10^{-6} for the first 4 epochs. Learning Rate Multipliers was used to control the adjustment of weights. In particular, to avoid local minimums, the upper layers were trained with a slower learning rate of 10^{-6}, while the last four layers with a faster learning rate of 10^{-2}. By using multiple learning rates, which were selected by experimentation, the weights of the top twelve layer went through some minor changes, while at the same time, the last four layers have converged faster.

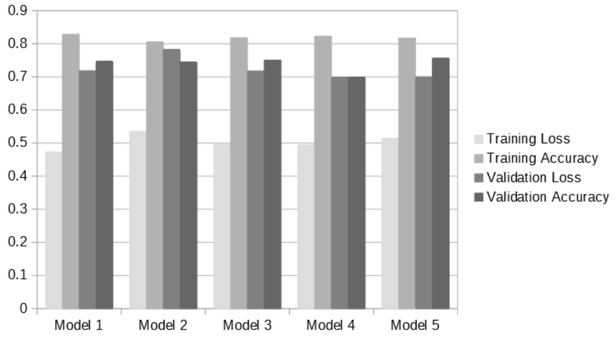
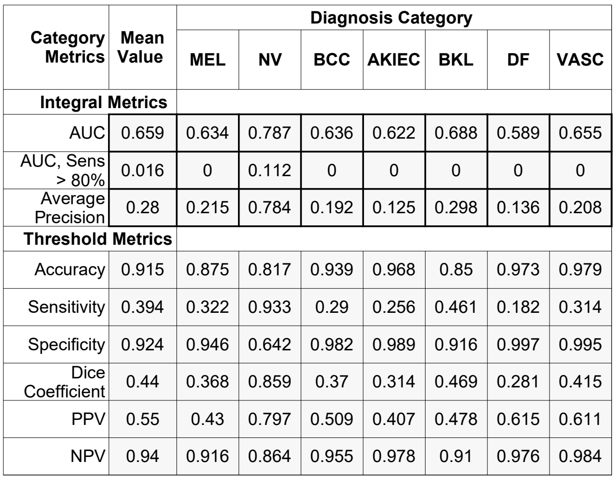


Figure 7. Final results from training and validation of each model.

D. Image Classification Results and Discussion

It should be taken into account that due to the limited computational power of the machine, the models were trained only for 30 epoch instead of 50 or 100 epochs. Five independently trained VGG16 models that learn essentially the same task, but converge to different solutions due to different learning parameter values and training dataset. The results of the various models obtained on the training and validation dataset are shown in Figure 7.

[2] https://challenge2018.isic-archive.com/task3/training/.



Category Metrics	Mean Value	Diagnosis Category						
		MEL	NV	BCC	AKIEC	BKL	DF	VASC
Integral Metrics								
AUC	0.659	0.634	0.787	0.636	0.622	0.688	0.589	0.655
AUC, Sens > 80%	0.016	0	0.112	0	0	0	0	0
Average Precision	0.28	0.215	0.784	0.192	0.125	0.298	0.136	0.208
Threshold Metrics								
Accuracy	0.915	0.875	0.817	0.939	0.968	0.85	0.973	0.979
Sensitivity	0.394	0.322	0.933	0.29	0.256	0.461	0.182	0.314
Specificity	0.924	0.946	0.642	0.982	0.989	0.916	0.997	0.995
Dice Coefficient	0.44	0.368	0.859	0.37	0.314	0.469	0.281	0.415
PPV	0.55	0.43	0.797	0.509	0.407	0.478	0.615	0.611
NPV	0.94	0.916	0.864	0.955	0.978	0.91	0.976	0.984

Figure 8. Overall score for majority vote from the five trained networks.

We have used a number of techniques to safeguard from overfitting, although the difference in the performance on the validation and challenge test datasets point to the limitation of the architecture to generalize across datasets and unseen images.

The results obtained by testing the networks on the dataset of “Task 3: Lesion Diagnosis” at the “ISIC 2018: Skin Lesion Analysis Towards Melanoma Detection” grand challenge test dataset are shown in Figure 9.

The best accuracy of 0.394 was achieved by using a majority vote on the outputs generated by the five different neural networks. Various metrics of the best model generated by the task leaderboard are presented in Figure 8.

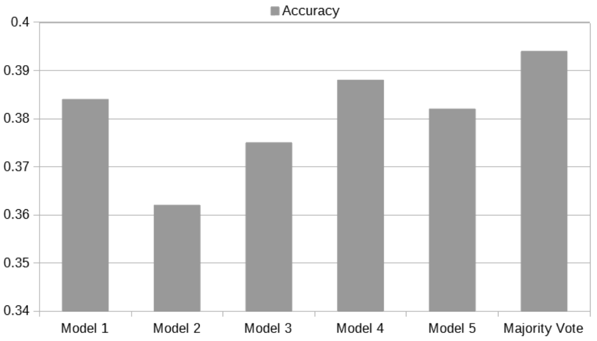


Figure 9. Achieved accuracy on TASK:3 LESION DIAGNOSIS test dataset.

IV. Conclusion

In this paper, a method for medical image segmentation based on image processing techniques have been proposed and evaluated on a grand challenge 2018 dataset. In the last decade, a remarkable progress had been made in image recognition mainly due to the advances in the field of deep learning. Convolutional neural networks have been applied to a number of diagnostic problems in dermatology, from distinguishing between various types of malignant and benign skin diseases to segmentation of skin lesions. We have also presented a deep learning architecture that builds upon a pre-trained VGG16 that was tailored for automatic melanoma detection that could support diagnosis from medical images in clinical dermatology.The accuracy results obtained by the teams participating in the challenge lie on a scale from 0.15 to around 0.80, although we should mentioned that the higher scores were obtained by using external resources, which is not the case in our experiments. Our results reflect the capability of the model to learn solely from the training data.

V. Acknowledgment

This work was partially financed by the Faculty of Computer Science and Engineering at the Ss. Cyril and Methodius University, Skopje, Macedonia.

References

[1] L. Queen, “Skin cancer: Causes, prevention, and treatment,” *School of Health Sciences*, 2017.

[2] A. Youssef, A. Pennisi, D. Bloisi, D. Nardi, M. Muscio, and A. Facchiano, “Deep convolutional pixel-wise labeling for skin lesion image segmentation,” 06 2018.

[3] A. Norouzi, M. Rahim, A. Altameem, T. Saba, A. Ehsani Rad, A. Rehman, and M. Uddin, “Medical image segmentation methods, algorithms, and applications,” *IETE Technical Review*, vol. 31, pp. 199–213, 06 2014.

[4] S. Srinivas, R. K. Sarvadevabhatla, K. R. Mopuri, N. Prabhu, S. S. S. Kruthiventi, and R. V. Babu, “A taxonomy of deep convolutional neural nets for computer vision,” *Frontiers in Robotics and AI*, vol. 2, p. 36, 2016.

[5] P. Tschandl, C. Rosendahl, and H. Kittler, “The ham10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions,” *Scientific Data*, vol. 5, p. 180161, Aug 2018.

[6] N. Codella, V. Rotemberg, P. Tschandl, M. E. Celebi, S. Dusza, D. Gutman, B. Helba, A. Kalloo, K. Liopyris, M. Marchetti, H. Kittler, and A. Halpern, “Skin lesion analysis toward melanoma detection 2018: A challenge hosted by the international skin imaging collaboration (isic),” 2019.

[7] J. A. Salido and C. Ruiz Jr, “Hair artifact removal and skin lesion segmentation of dermoscopy images,” *Asian Journal of Pharmaceutical and Clinical Research*, vol. 11, p. 36, 10 2018.

[8] B. Lin, K. Michael, S. Kalra, and H. Tizhoosh, “Skin lesion segmentation: U-nets versus clustering,” pp. 1–7, 11 2017.

[9] A. Telea, “An image inpainting technique based on the fast marching method,” *Journal of Graphics Tools*, vol. 9, no. 1, pp. 23–34, 2004.

[10] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *CoRR*, vol. abs/1409.1556, 2014.

[11] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *arXiv preprint arXiv:1409.1556*, 2014.

[12] A. Romero Lopez, X. Giro-i-Nieto, J. Burdick, and O. Marques, “Skin lesion classification from dermoscopic images using deep learning techniques,” in *2017 13th IASTED International Conference on Biomedical Engineering (BioMed)*, pp. 49–54, Feb 2017.

[13] M. Stone, “Cross-validatory choice and assessment of statistical predictions,” *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 36, no. 2, pp. 111–133.

[14] J. D. Rodriguez, A. Perez, and J. A. Lozano, “Sensitivity analysis of k-fold cross validation in prediction error estimation,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, pp. 569–575, March 2010.

[15] D. Anguita, L. Ghelardoni, A. Ghio, L. Oneto, and S. Ridella, “The’k’in k-fold cross validation.,” in *ESANN*, 2012.

Framework for Recommending Clothes Based On Convolutional Network Derived Stylespaces

Stefan Kanan
Faculty of Computer Science and Engineering
Ss. Cyril and Methodius
Skopje, Macedonia
kanan.stefan@students.finki.ukim.mk

Ilinka Ivanoska
Faculty of Computer Science and Engineering
Ss. Cyril and Methodius
Skopje, Macedonia
ilinka.ivanoska@finki.ukim.mk

***Abstract*—Over the years the internet has taken a very central role in our society. While it might have been unthinkable in the past, many people today elect to read their news, do their shopping or subscribe to various services online. This has made it increasingly more important for sellers to offer the right ad, relevant to our interests. In this paper we investigate a framework for matching clothing products based on their stylistic compatibility. The algorithm works by using Convolutional neural networks to find an appropriate embedding, that is close to other stylistically compatible products. We create a framework in order to test the efficacy of this algorithm. Our results show that the framework is capable of recommending matching clothes better than those chosen at random. We also visualize our items using tSNE and compare our plot with that of the original paper.**

***Index Terms*—Siamese Networks, tSNE, Recommendation algorithm, Neural Networks**

I. Introduction

Today, more than ever advertisements are an unavoidable part of daily life. Whether reading an article, watching a video or simply opening an app on our smartphone, it seems like our gaze simply cannot escape the ubiquitous ads. Given our increasing fatigue, it is ever more important that advertisers wisely choose what to offer lest they lose our fleeting interest. This has brought in the age of tailored advertisement. Whether entertainment, vacation spots, hotels or clothes, large and complex algorithms are hard at work crunching data, to offer us suitable recommendations.

In this paper we implement the recommendation framework described in [6], which uses a Siamese network in order to pair items based on their compatibility. To that end a database is build, in which we store our items and a pipeline for recommending suitable clothes given some prior user specified clothing item is created. We conduct a user study where users have to choose whether the clothes they have been recommended by the algorithm are in fact a suitable match and see whether the algorithm is better than choosing items at random.

The algorithm works by translating the image of the apparel into a new style space with the aid of a neural network. This network is first created by supplying it with both compatible and incompatible pairs of clothes, with the hope that it will learn some notion of style with which it would be able to discern matching and non-matching items. This new style space brings items that go well together close to each other and splays items that do not further apart. Thus, for any given image, the algorithm can recommend an item it sees as the most compatible to the one given by the user.

The neural network assigns an embedding to each item, which describes its location in this new style space. Naturally items that go well together are expected to be close to each other and vice versa. Knowing this it would be beneficial to visualize the style space. However, the high dimensionality of the embeddings makes this unfeasible using traditional techniques. One way in which to express high dimensional data is using he tSNE visualization algorithm, which we will use here, that expresses high dimensional data in terms of the joint probability between nodes.

Our paper is structured as follows. In Section II we describe our dataset and all of the pre-processing steps as well as introduce the concept of the Siamese convolutional networks. In Section III we offer an implementation of [6] which is explained in detail. In section IV the tSNE algorithm is explained and we present the results of this algorithm on our data. In Section V we conduct a user study and discuss the results. Finally, we offer a conclusion.

II. Materials and Methods

A. Dataset

Our data consists of clothing products. Each product a contains: an id and a link to the image of a: I_a, additionaly some items contain a $description$. Part of the recommendation algorithm is matching an item to a target category (wristbands, dresses and etc). Our dataset doesn't contain a dedicated category attribute, yet the description part contains valuable information about the category the item belongs to. In order to extract the descriptive attribute into its corresponding categories, we first clean and tokenize this attribute for each item in our dataset. After doing so we count the appearance of each category in our dataset, then we pick manually the more commonly found categories while excluding artifacts, brands and other non-valid categories. Finally, we end up with a handful of basic clothing categories (such as pants, shirts and jackets). A caveat concerning our descriptors, is that in parsing this attribute rather than having a dedicated $category$ attribute we sometimes end up with the wrong categories for our items (ex. a belt whose description may say ”belt for jeans” but which because of the parsing process may end up in both the belt and the jeans category). In order to avoid this problem we suggest a more intelligent parsing algorithm or manually classifying items into their suitable categories. Another problem we ran into with our dataset, was duplicated data. We removed these duplicates on the basis of having the same image, however in doing so we ended with 7495 items from our original 26375 ones, this is in comparison with the nearly two hundred thousand images in [6].

B. *Methodology review*

The main idea behind [6] is the use of Convolutional neural networks to learn a style space so that clothes that stylistically go well together are part of the same style space, while clothes that are not compatible, are not.

This is done by utilizing a Siamese neural network, trained on mini-batches of 1 compatible pair of clothes $(a, b)^+$ and 16 negative pairs $(a, b)^-$ [6]. These pairs are heterogeneous dyads, which means that the two items come from different clothing categories. In doing so, they argue, we force the network to learn to match items from different categories, rather than group similar items together.

The Siamese neural networks are a specie of Convolutional neural networks consisting of two identical neural networks sharing weights θ. During training the neural networks are fed a pair of items and the desired output y, denoting whether a pair is compatible or not. The result is then used in a contrastive loss function [1], which calculates the error and is used to train the network:

$$L(\theta) = \sum_{x_q, x_p} L_p(x_q, x_p) + \sum_{x_q, x_n} L_n(x_q, x_n) \qquad (1)$$

Where the first term penalizes compatible images that are found to be far apart and the second term gives the penalty for incompatible images that are nearby.

$$L_p(x_q, x_p) = \|x_q - x_p\|_2^2 \qquad (2)$$

$$L_n(x_q, x_n) = \max(0, m^2 - \|x_q - x_n\|_2^2) \qquad (3)$$

In our case, given two items and their images $a : I_a$ and $b : I_b$ we want to translate these images into their stylespace or rather find their embeddings $F : I_a \rightarrow s_a$ and $F : I_b \rightarrow s_b$. This comes down to finding suitable descriptors which are either distant if the two items are not compatible or close to each other if they are. [6] uses a GoogleNet network, pre-trained on the ILSVRC image dataset [4]. The original intention of which was to correctly classify images. The Siamese network is then build by removing the last softmax layer, and replacing it with a 256 node fully connected layer. [6] [1]

The network is now fine-tuned using pairs from different categories in order to force the network to learn style compatibility rather than classifying items as near when they are from the same category. These pairs make use of the ’items frequently brought together’ section on the Amazon dataset. [2] [3]

Finally, in order to recommend an outfit the algorithm clusters items from the same category. In the paper this is done using the k-means algorithm with 20 clusters per category in order to lessen the calculation when comparing items. Then our query item finds the nearest centroid and is compared with items in that cluster. Finally, we use a knn algorithm with $n = 5$ to find the 5 closest items and from them choose the closest one to show.

III. Implementation

Our implementation uses the pre-trained network[1] in [6]. We also build a database in which to store our items and embeddings. Because each item a can be part of many categories $a \in C_i$ where $i = 1...n$ and many clusters, we deemed that a relational database would prove unsuitable given the structure of the data, and would provide worse performances as the amount of items increased. Therefore we elected to use Neo4j, a graph database that stores data in the form of nodes (which can have one or more roles), edges (representing relationships) and attributes. Neo4j has the added benefit of constant performances regardless of the amount of data. A graph database is uniquely equipped to deal with items that can be part of many categories whilst keeping the model clean resulting in simple queries. A model of our database can be seen in (fig 1). Next we run our Siamese network to extract the embeddings from our images, using each category and gender we then cluster our items. Our implementation tends to keep the recommended 20 clusters per category, except for less numerous categories where we use 10 clusters, and for categories with 50 items or less which are not clustered.

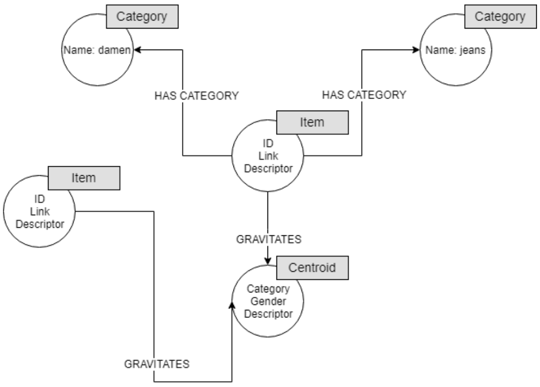


Fig. 1. The Neo4j model

Finally, we build a pipeline where a user can provide either an id or a link and a target category and gender and can be recommended compatible items. We do this by using a knn algorithm which finds the 5 most compatible items. Even

[1] The network, implemented in Caffe, can be found at https://vision.cornell.edu/se3/projects/clothing-style/

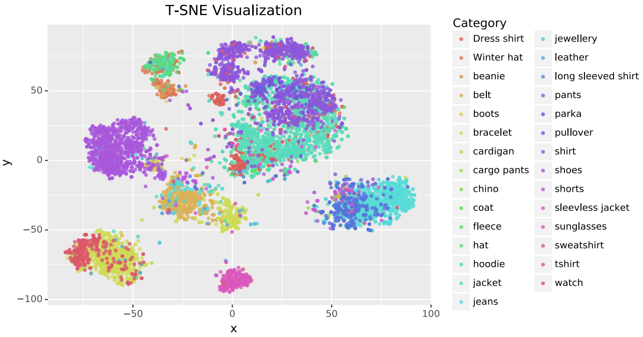


Fig. 2. Visualization of the item stylespace using tSNE, $Perplexity = 50$, 5000 iterations

though [6] recommends clustering the items and finding the knn for our item in part to avoid presenting the user with items from the same category as our query item that have been miscategorized, we found that this was frequently the case in our pipeline. We hypothesize that the network still classifies items from the same category as being similar, perhaps a different training regiment may solve this issue, however more research is needed on this topic.

IV. T-SNE Visualization

As in the original paper, after finding the descriptors for all of our items, we visualize them using the tSNE algorithm. tSNE (Short for 't-Distributed Stochastic Network Embedding') is a technique for visualizing highly dimensional data which works by converting the original data $x_i \in X$ for $i = 1, 2...n$ (each x_i is a row of variables in some dataset X) into two or three dimensional coordinates fit for visualizing, this latter set is labeled as $y_i \in Y$ for $i = 1, 2...n$, called a map. [5] The plain SNE algorithm first converts the euclidean distance between points into conditional probabilities, and then tries to find lower dimensional points whose conditional probability matches the one on the high dimensional data. The conditional probability of x_i is

$$p_{i|j} = \frac{\exp\left(\frac{-\|x_i - x_j\|^2}{2\sigma_i^2}\right)}{\sum_{k \neq j} \exp\left(\frac{-\|x_i - x_j\|^2}{2\sigma_i^2}\right)} \tag{4}$$

.

With an unknown variance σ_i^2, the variance is found by using

$$Perp(P_i) = 2^{H(P_i)} \tag{5}$$

Where $Perp$ is the perplexity or the approximate number of neighbours of each node, and is given by the user, and $H(P_i)$ is the conditional entropy

$$H(P_i) = -\sum_j p_{j|i} \log_2 p_{j|i} \tag{6}$$

.

Correspondingly, the conditional probability of the lower dimensional representation of our data, is

$$p_{i|j} = \frac{\exp\left(-\|y_i - y_j\|^2\right)}{\sum_{k \neq j} \exp\left(-\|y_i - y_j\|^2\right)} \tag{7}$$

with a fixed variance of $\sigma^2 = \frac{1}{2}$, this scales the map, but in doing so loses some detail of the original data, for there, every point had its own variance. In order to find suitable lower dimensional data points, we minimize the Kullback-Leibler divergence

$$C = \sum_i \sum_j p_{j|i} log \frac{p_{j|i}}{q_{j|i}} \tag{8}$$

The cost function is then minimized by using the gradient descent method. However this cost function of the plain SNE algorithm can often times end up in a local minima, requiring multiple runs or other optimization. Because of the way the Kullback-Leibler divergence is laid out, the equation will favour mapping the points in the vicinity of the node, no matter their distance. Moreover, because the algorithm maps higher dimensional data to a lower dimensional, there's an inherent problem of crowding. To avoid these issues the tSNE algorithm uses the joint probability rather than the conditional probability, where the joint probability of the higher dimensional data is defined as

$$p_{ij} = \frac{p_{i|j} + p_{j|i}}{2n} \tag{9}$$

.

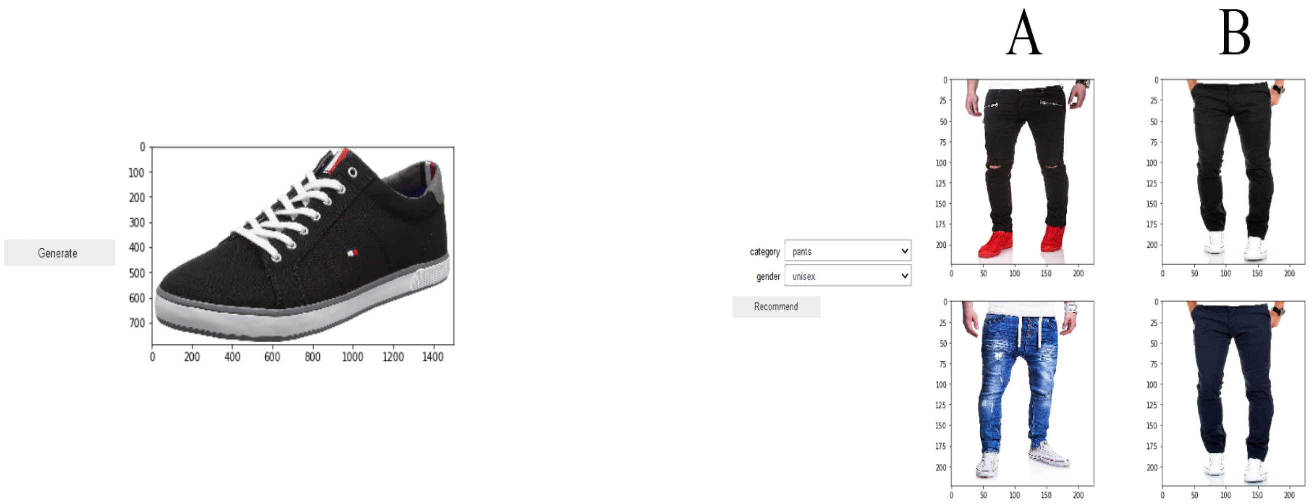
As for the joint probability in the lower dimensional representation, rather than a Gaussian distribution, tSNE uses a one degree of freedom Student t-Distribution (Cauchy).

$$q_{ij} = \frac{(1 + \|y_i - y_j\|^2)^{-1}}{\sum_{k \neq l}(1 + \|y_i - y_j\|^2)^{-1}} \tag{10}$$

.

Because of the heavier tail, it spreads the points outwards, therefore avoiding the cluttering of points that should be further apart. These newly defined joint probability also lessen the cost of the optimization.

Since the original paper, gives no details about the setting of the perplexity; Several different values were tried, as different



category pants
gender unisex

Fig. 3. Users press the 'Generate' button to get an image at random, next they select the type of clothes they want recommended and whether they want them to be male, female or unisex. Pressing the 'Recommend' button produces two columns one of which is chosen at random and the other with the help of our algorithm, the user then picks the column which better suits the formerly generated image

perplexities can have vastly different results, each showing a different aspect of the global or local structure of the data. [7]. We ran the algorithm for five thousand iterations after which we plotted the results. As is indicated in the paper, items of the same category cluster together, however while in the original paper, the different clusters were tightly woven with each other, in our results, while there is considerable overlap between categories, some categories formed groups of their own and were rather separated from each other. Whether this is simply because of the data or just an negligible intricacy of the tSNE algorithm is unclear. Though distances in tSNE may not have much meaning and in reality could only be a moderate distance away. [7] [5] While the tSNE algorithm did not converge for any of these values, this result persisted for different perplexity values ($Perp = 2, 10, 30, 50, 100, 200$). The tSNE visualization can be seen in (fig 4).

V. User Study

In order to test the effectiveness of the algorithm in [6] we conduct a user study. Users were given an image picked at random, next they were asked to select the type of garments they wished recommended to them based on the previously generated image. Two columns were generated, one of which was chosen at random and the other with the help of our algorithm, the surveyed users then picked the column which they thought better suited the formerly generated image. Therefore users couldn't distinguish between picking from the random column or from the algorithm one. We finally collected and compared the results of our survey.

During our survey, some users found several category names to be mistranslated, this could have affected their choice. Confusion could also have been averted by making the user interface more intuitive. Another problem was that due to the miscategorized items, our framework used the results from the third, fourth or fifth cluster rather than the first one as suggested in the paper. We suspect that we would have obtained better results were we to use the latter. Even so, our survey shows promising results. With 90 votes for the recommended items opposite the 60 votes for the random column, this result is statistically significant with a 95% confidence level (fig. 2).

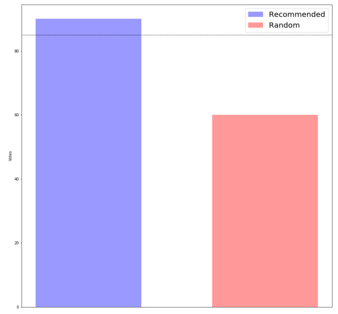


Fig. 4. Results from the user study. The dotted line indicates the cutoff for the statistically significant results (Obtained using a one tailed binomial distribution test, 95% confidence).

VI. Conclusion

This paper describes an implementation of the concepts and framework first introduced in [6]. Using their pretrained Siamese network we were able to get the descriptors of our items. We built a database and a pipeline for clustering and recommending suitable pairs of clothes. We conducted a user study in which we asked users to rate whether or not the recommended items were suitable or not, thus testing the performances of this algorithm. Finally, we visualized our data using the tSNE technique and noticed a difference with the plot in the original work. Further work can investigate the difference between our visualization and the one in the original paper, as well as focus on better fine-tuning the network or extending this approach to other fields like music, shopping and etc.

REFERENCES

[1] Sean Bell and Kavita Bala. Learning visual similarity for product design with convolutional neural networks. *ACM Trans. Graph.*, 34(4):98:1–98:10, July 2015.

[2] Ruining He and Julian McAuley. Ups and Downs: Modeling the Visual Evolution of Fashion Trends with One-Class Collaborative Filtering. In *Proceedings of the 25th International Conference on World Wide Web - WWW '16*, pages 507–517, Montr&#233;al, Qu&#233;bec, Canada, 2016. ACM Press.

[3] Julian McAuley, Christopher Targett, Qinfeng Shi, and Anton van den Hengel. Image-based Recommendations on Styles and Substitutes. *arXiv:1506.04757 [cs]*, June 2015. arXiv: 1506.04757.

[4] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Computer Vision and Pattern Recognition (CVPR)*, 2015.

[5] Laurens van der Maaten and Geoffrey E. Hinton. Visualizing data using t-sne. 2008.

[6] Andreas Veit*, Balazs Kovacs*, Sean Bell, Julian McAuley, Kavita Bala, and Serge Belongie. Learning visual clothing style with heterogeneous dyadic co-occurrences. In *International Conference on Computer Vision (ICCV)*, Santiago, Chile, 2015. *The first two authors contributed equally.

[7] Martin Wattenberg, Fernanda Vigas, and Ian Johnson. How to use t-sne effectively. *Distill*, 2016.

Regression Models Comparison on Elo Merchant Category Recommendation Kaggle Competition Data Set

Jane Lameski, Dejan Gjorgjevikj
Faculty of Computer Science and Engineering
Ss.Cyril and Methodius University, Skopje, Macedonia
Email: jane.lameski@students.finki.ukim.mk, dejan.gjorgjevikj@finki.ukim.mk

Abstract—**Regression, as a tool widely used for prediction and forecasting, its use has substantial overlap with the field of pattern recognition and machine learning. The regression analysis methods and their performance, in practice, depends on the form of the data, and how it relates to the regression approach being used. In this paper, we have compared the success of several state-of-the-art regression models, and general regression techniques such as the Linear regression.**

Index terms— **Pattern recognition, machine learning, regression**

I. Introduction

Regression derives from statistical measurement used in finance, investing and other disciplines which measure the relationship between one dependent variable (label) and series of other changing variables. Regression is helpful statistic method that can be leveraged across an organization to determine the degree to which concrete independent variables are influencing dependent variables.

The possible scenarios for conducting regression analysis to yield valuable, actionable business insights, or computing optimal approximation to a label for a given problem and data set are endless.

From problem to problem, the power of ones model most probably will differ, meaning, for the purpose of comparison different regression models, in this research we used data set from Kaggle competition ”Elo Merchant Category Recommendation”[1].

This competition is sponsored by Brazil’s leading bank [2]. It has machine learning models built to understand the most important preferences and aspects in their customers’ lifecycle. Because of none existing tailor for individuals or profiles, the competitors by uncovering signal in customer loyalty are supposed to serve and identify the most relevant opportunities to individuals.

The paper is organized as follows. In the following section, the models with the best results are described. The third section, the results are shown from the models tested on the data sets. In the forth section, the data sets are explained, as well as the changes made on them and how they were used.

II. Models

The research will further discuss and compare the results obtained from the competition as a guide, which shows which model would most probably show most valuable results according to the data set which we used. In the following parts of this section, all the models used, had no parameter tuning, which is a reason for the bad results shown by some of them.

A. Linear Regression

The two basic types of regression are linear regression and multiple linear regression. The Linear regression is defined by the following formula:

$$Y = a + bX + r$$

Similarly, the multiple linear regression is defined as:

$$Y = a + b_1X_1 + b_2X_2 + ... + b_nX_n + r$$

where Y is the variable we try to predict or dependent variable (label), X_i the variables we use to predict Y, b_i the slope, a the intercept with the y-axis in the Decart Coordinate System (DCS) and r is the regression reminder. What regression does, is it takes a group of random variables (in our case X_i, and tries to find mathematical relationship between them (typically a straight line) that best approximates all the individual data points.

B. ElasticNet Regression

ElasticNet [3] is a hybrid of two regression techniques the Lasso [4] and Ridge [5] regression. This model linearly combines the L_1 and L_2 regularization penalties taking on the effects from both of the techniques:

$$min(||Xw - y||^2 + L_1|w| + L_2|w|^2)$$

where $X = (x_1|...|x_p)$ is the model matrix, $y = (y_1, ..., y_n)^T$ is the response and $L_1|w|$ and $L_2|w|^2$ are the two penalty parameters defined as:

$$L_1|w| = \sum_{j=1}^{p} |w_j|$$

$$L_2|w|^2 = \sum_{j=1}^{p} w_j^2$$

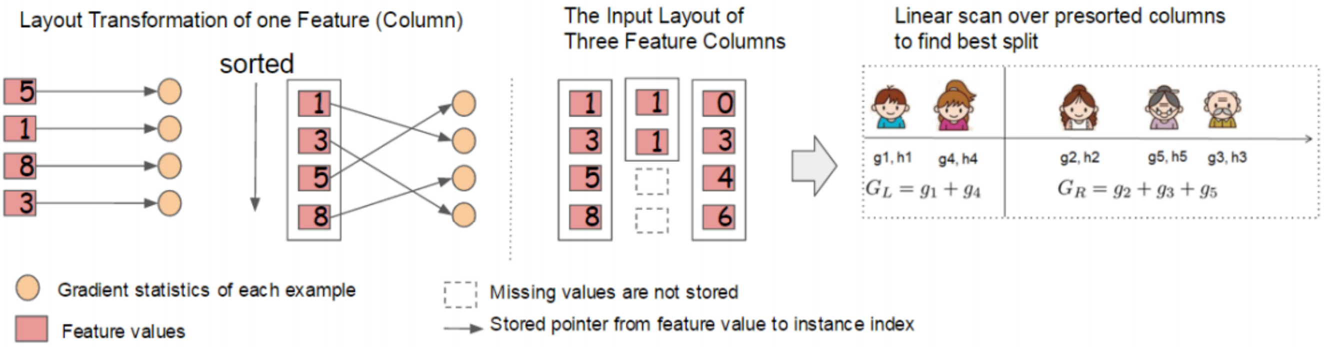

Layout Transformation of one Feature (Column)
sorted
5
1
8
3
1
3
5
8
The Input Layout of
Three Feature Columns
1
3
5
8
1
1
0
3
4
6
Linear scan over presorted columns
to find best split
g1, h1
g4, h4
g2, h2
g5, h5
g3, h3
$G_L = g_1 + g_4$
$G_R = g_2 + g_3 + g_5$
Gradient statistics of each example
Feature values
Missing values are not stored
Stored pointer from feature value to instance index

Figure 1: Block structure for parallel learning. The columns in a block are sorted by the feature values, and later a linear scan over one column in the block is sufficient to enumerate all the split points.

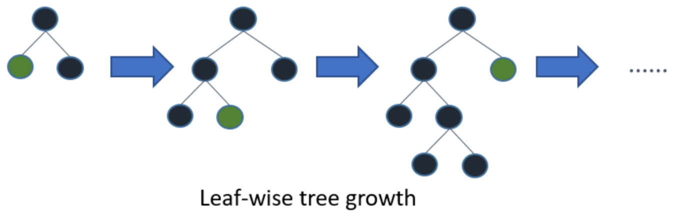


Figure 2: Leaf wise tree growth.

A particular advantage of the trading-off between Lasso and Ridge is that it allows ElasticNet to inherit some of Ridges's stability under rotation.

ElasticNet encourages group effect in case of highly correlated variables, rather than zeroing some of them like Lasso method does and there are no limitations to the number of selected variables. This model was used because of its well results in case of high dimensionality and multicollinearity among the variables in the data set

C. Extreme Gradient Boosting Regression

Extreme Gradient Boosting [6], is one of the most popular end-to-end tree boosting machine learning method, achieving state-of-the-art results on numerous machine learning challenges. This algorithm is a tree learning algorithm, handling sparse data. It also exploits out-of-score computations and enables hundred millions of examples to be processed on a desktop in meaningful time. This is achieved thanks to its parallel learning. In other words, because the slowest part of tree learning is to sort the data, the data is stored in in-memory units called blocks in compressed column (CSC) format. The transformation of the data set into the format and finding the optimal split candidates in the leaf branches using the block structure is shown in Fig: 1

D. Light Gradient Boosting Machine

Microsoft's Light Gradient Boosting Machine, as stated in [7], *contains two novel techniques: Gradient-based One-Side Sampling and Exclusive Feature Bundling to deal with large number of data instances and large number of features respectively*. Which is exactly what we need after the merge of the given data sets from the competition, and further expansion with the feature aggregations and combinations. The idea is when creating new leaves, instead of checking all of the splits, only some are checked. First, all the attributes are sorted and bucket the observation by creating discrete bins. Instead of iterating over all leaves when we want to split a leaf in the tree, we simply iterate over the buckets. This method is named histogram implementation by the authors of the paper. The trees are grown with leaf wise strategy [8] as shown in Fig: 2, having the presorted state instead of level wise like other gradient boosting methods.

E. Random Forests

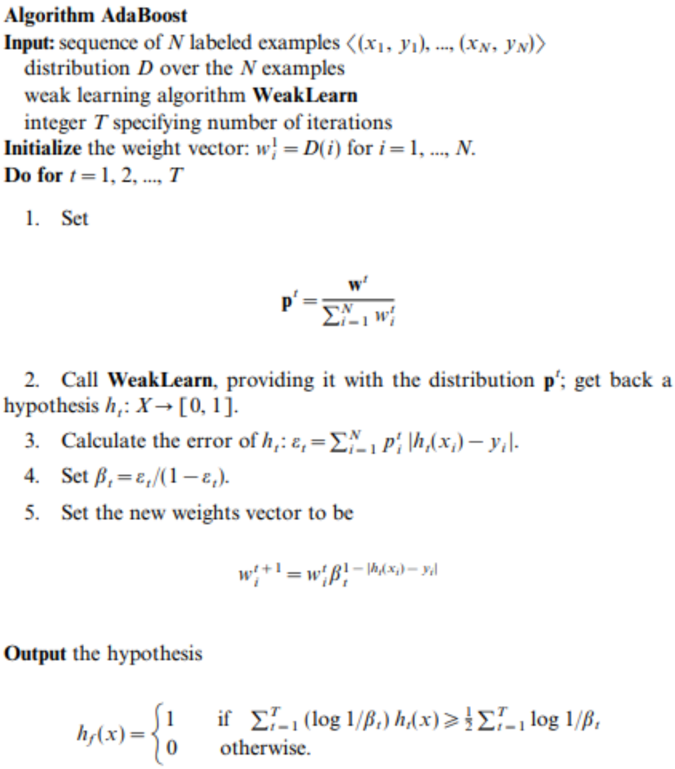
Random forests [9] adds an additional randomness layer to bagging [10]. Additionally, in the construction of each tree, different bootstrap sample of data is used. In standard trees best split among variables in used, in random forest, each node is split using the best from randomly chosen predictors at that node.

F. AdaBoost Regression

AdaBoost or Adaptive Boosting Algorithm [11] is an algorithm that adjusts adaptively to errors of weak hypotheses returned by Weak Learner. AdaBoost, unlike boost-by-majority, combines this weak hypotheses by summing their probabilistic predictions. The algorithm itself, is shown with the pseudo code in Fig: 3

G. Bagging

Bagging [10] is a method that creates ensembles by ”bootstrap aggregation” with repeatedly randomly resampling the training data. It trains M learners on M learners on M bootstrap samples and combines the outputs by voting (majority vote). This method decreases the variance in the results due to unstable learners, algorithms that produce output which can dramatically change when training data is slightly changed.



```
Algorithm AdaBoost
Input: sequence of $N$ labeled examples $\langle(x_1, y_1), ..., (x_N, y_N)\rangle$
    distribution $D$ over the $N$ examples
    weak learning algorithm WeakLearn
    integer $T$ specifying number of iterations
Initialize the weight vector: $w_i^1 = D(i)$ for $i = 1, ..., N$.
Do for $t = 1, 2, ..., T$
    1. Set
           $\mathbf{p}^t = \frac{\mathbf{w}^t}{\sum_{i=1}^{N} w_i^t}$
    2. Call WeakLearn, providing it with the distribution $\mathbf{p}^t$; get back a hypothesis $h_t: X \to [0, 1]$.
    3. Calculate the error of $h_t$: $\varepsilon_t = \sum_{i=1}^{N} p_i^t |h_t(x_i) - y_i|$.
    4. Set $\beta_t = \varepsilon_t/(1 - \varepsilon_t)$.
    5. Set the new weights vector to be
           $w_i^{t+1} = w_i^t \beta_t^{1 - |h_t(x_i) - y_i|}$
Output the hypothesis
    $h_f(x) = \begin{cases} 1 & \text{if } \sum_{t=1}^{T} (\log 1/\beta_t) h_t(x) \geq \frac{1}{2} \sum_{t=1}^{T} \log 1/\beta_t \\ 0 & \text{otherwise.} \end{cases}$
```

Figure 3: Adaptive Boosting Algorithm.

III. Results

For all the models mentioned in section II their training and test time performance can be seen below in Table:I, and their scores achieved from the [1] are shown in Table:II. The scores are obtained by root mean square error(RMSE) defined as:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2},$$

where $\hat{y}$ is the predicted loyalty score for each card id, and y is the actual loyalty score assigned to a card id.

Table I: Training and Testing time of the used models

Regression model name	Training time	Testing time
Linear	**1.49 sec**	**0.16 sec**
XGBoost	76.95 sec	0.76 sec
LightGBM	5.18 sec	0.66 sec
RandomForest	276.41 sec	0.80 sec
Bagging	278.78 sec	2.12 sec
AdaBoost	175.41 sec	0.83 sec
ElasticNet	15.29 sec	0.17 sec

IV. Data Set

For this study, we use the data set from the competition Elo Merchant Category Recommendation, which goal is to predict a customer loyalty score. There are multiple data tables given by the competition, train, test, historical transactions and new merchant transactions and merchants. The train file consists of 201917 rows and 6 columns which are card ids which are unique card identifiers, first active month or the month of first purchase, three anonymised categorical features and a target vale which is a numerical score calculated two months after historical and evaluation period. The test file has the same features as the train file, but it has 123623 rows. The historical and new merchant transactions files have same features, historical transactions has 29112361 rows and 14 columns. It contains up to 3 months' worth of transactions for every card at any of the provided merchant ids. New merchant transactions has 1963031 rows and 14 columns. It contains the transactions at new merchants (merchant ids that this particular card id has not yet visited) over a period of two months. Out of this 14 features 6 are ids, 4 are categorical and 4 are numerical. These data sets consist of card ids same as in train and test, month lag to reference date, date of purchase, authorized flag, three other anonymised categories, number of installments of purchase, merchant category, merchant category group and merchant identifiers, normalized purchase amount and city and state identifier. The historical and new merchant transactions are meant to be merged with the train and test data sets. These sets were further expanded by adding several features such as month difference between today's date and date of purchase added to the month lag, day difference, day difference between one's transactions, the month of purchase and the ratio between the number of transactions in new merchant transactions and historical transactions. To merge this data sets, we used aggregation of the features. At first, the merge was done under almost all statistical measures, such as mean, minimum, maximum, standard deviation, variation, standard error of the mean and sum. The final train set consists of same number of rows with 137 columns, including the target, and the final test set has 136 columns. The missing values in the data sets are replaced with the mean value for that particular feature. On Fig: 4a we show target points which are far from the mean value of the column when plotted against feature_2 and feature_3. On Fig: 4b and Fig: 4c we see purchase_amount and installments values plotted against each of the categories. From here, the fact that there are few values which are outliers, we change these values with the mean of the corresponding feature.

Table II: Scores obtained from the regression models

Regression model name	Kaggle Private Score	Kaggle Public Score
Linear	3.81507	3.93070
XGBoost	**3.67364**	3.76776
LightGBM	3.68492	**3.76112**
RandomForest	5.46625	5.50059
Bagging	5.88832	5.88293
AdaBoost	6.83771	6.80986
ElasticNet	4.35340	4.44727

V. Conclusion

A widely used tool for prediction and forecasting, the regression, as a statistical tool, is studied and researched in the fields of pattern recognition and machine learning. The regression analysis methods and their outcome, in practice, depends on the form of the data, and how it relates to the regression approach being used. With the comparison shown in the results of several state-of-the-art regression models we can conclude that Extreme Gradient Boosting and Light Gradient Boosting Machine are the most promising models

VI. Acknowledgement

This research work was conducted as a student project for the undergraduate course ”Pattern Recognition” at the Faculty of Computer Science and Engineering, Saints Cyril and Methodius University, Skopje.

References

[1] “Elo Merchant Category Recommendation.” https://www.kaggle.com/c/elo-merchant-category-recommendation. Accessed: 2019-03-06.

[2] “Elo Official Web Site.” https://www.elo.com.br. Accessed: 2019-03-06.

[3] H. Zou and T. Hastie, “Regularization and variable selection via the elastic net,” *Journal of the royal statistical society: series B (statistical methodology)*, vol. 67, no. 2, pp. 301–320, 2005.

[4] R. Tibshirani, “Regression shrinkage and selection via the lasso,” *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 58, no. 1, pp. 267–288, 1996.

[5] A. E. Hoerl and R. W. Kennard, “Ridge regression: Biased estimation for nonorthogonal problems,” *Technometrics*, vol. 12, no. 1, pp. 55–67, 1970.

[6] T. Chen and C. Guestrin, “Xgboost,” *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD ’16*, 2016.

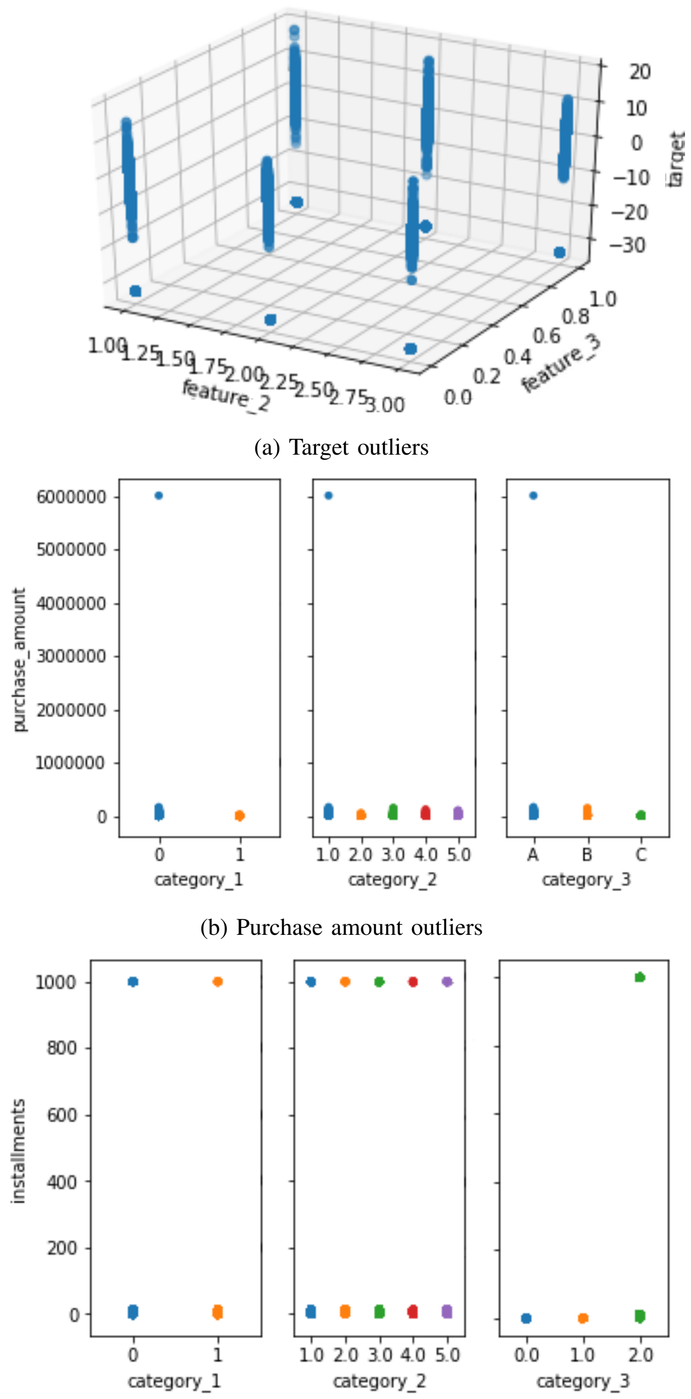
[7] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu, “Lightgbm: A highly efficient gradient boosting decision tree,” in *Advances in Neural Information Processing Systems*, pp. 3146–3154, 2017.

[8] H. Shi, *Best-first decision tree learning*. PhD thesis, The University of Waikato, 2007.

[9] L. Breiman, “Random forests,” *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.

[10] L. Breiman, “Bagging predictors,” *Machine learning*, vol. 24, no. 2, pp. 123–140, 1996.

[11] Y. Freund and R. E. Schapire, “A decision-theoretic generalization of on-line learning and an application to boosting,” *Journal of computer and system sciences*, vol. 55, no. 1, pp. 119–139, 1997.



(a) Target outliers

(b) Purchase amount outliers

(c) Installments outliers

Figure 4: Plots of relation between Target and feature 1 and 2, Purchase amount and categories 1,2 and 3, and installments with category 1, 2 and 3

Riemann Garbage Bin: A Self-sustained Waste Management System

Kiril Zelenkovski
Faculty of Computer Science and Egineering
Skopje, Macedonia
kiril.zelenkovski@students.finki.ukim.mk

Filip Karafiloski
Faculty of Computer Science and Egineering
Skopje, Macedonia
filip.karafiloski@students.finki.ukim.mk

Igor Mishkovski
Faculty of Computer Science and Egineering
Skopje, Macedonia
igor.mishkovski@finki.ukim.mk

Abstract: This paper elaborates the idea for building a self-sustained sensor system, a network composed of nodes implementing the Riemann Garbage Bin (RGB) model. Hence, the case study explores the potential of employing sensor enabled systems to improve on waste monitoring and management in public waste bins. The network consists of wireless nodes that use ultrasonic sensors to measure the empty space in the bins. The sensors periodically report the fill rate of the waste bins to a sensor gateway that is based on Long Rage Wide Area Network (LoRaWAN) protocol. For our LoRaWAN server and network cover we choose to use The Things Network (TTN). These fill rates would be monitored by a mobile or web application connected to the network server. The goal of this project is through the Internet of Things (IoT) to monitor all the waste bins in one city, to improve the garbage management by relocating resources and by giving insight to the public about this global health threating problem.

Keywords – sensors, LoRaWAN, TTN, IoT

I. Introduction

During the last century the world population has been rising, and there has been a major migration from rural to urban areas. Today 50% of the world's population inhabit cities and this number is expected to reach 70% by 2050 [1]. As this global problem of migration shift towards the urban areas, the capital of the Republic of N. Macedonia, Skopje, also faces these complex problems like pollution, traffic and waste management. In parallel, the recent years have witnessed the rise of the 'smart cities', where the governments are challenged [2, 3] to tackle this health threatening problems with technology driven solutions. According to the Public Company Communal Hygiene Skopje the number routes for waste collection in 2018 has dropped for 2% in comparison to 2017 statistics. Hence, the amount of fuel used should also drop. But this is not the case. Fuel consumption int 2018 has raised for 3,3%. This number indicates that the trucks that collect waste are only suffocating the traffic flow and are causing more damage to the environment. Traditionally, waste collection has been preformed on a fixed schedule, which is the case in Skopje. However, regular schedule is not optimal as it does not account that different areas fill up their bins in different rates. This means collection trucks must stop at each point on their route to empty waste bin regardless of whether they are full or not. It also leads to situations where some waste bins overflow before the next collection schedule. Through our model we intend to provide a solution that would reduce the operational costs by streamlining their routes to deploying waste bins only where it is necessary.

II. Model Description

The RGB model is consisted of multiple components:

A. Microcontroller

The market for microcontrollers grew almost 10% from 2017 to 2018 [4]. This fast growth has made the Arduino microcontrollers and all sorts of different sensors highly available. Anyone with a simple idea and few dollars on their credit cards, can go online to Amazon or AliExpress and order these small devices. Although this availability and simple structure makes them easy to integrate in any IoT solutions, they hardly find direct usage in the industrial sector. Our solution focuses on building a low-cost IoT waste management system. Furthermore, Figure 1 shows the microcontroller that we use for controlling our sensors [5]. It is the Arduino Mega2560 board, by ELEGOO A high-performance 8-bit AVR RISC-based microcontroller combines 256KB ISP flash memory.

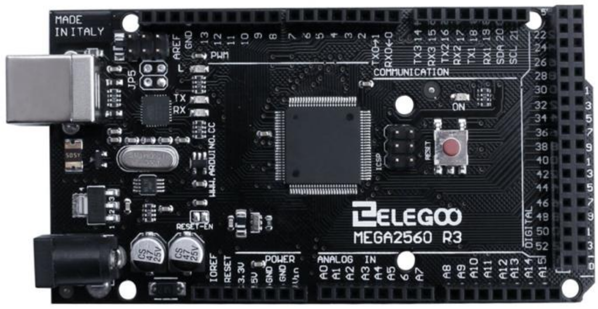


Fig. 1. Arduino Mega2560 Board [5]

B. Opening system

The constant unbearable smell cased by opened waste bins in Skopje has made us develop a certain type of opening system for bins that have lids. Although this solution is maybe inappropriate for implementing on bins that don't have lids, we made our protype model does have a lid. It uses one servo motor, which is triggered to open by the following three actions:

- RFID cards – these will be used employees from the Communal Hygiene
- Fire – if there is a fire detected the lid will open, making it easier for putting the fire out

- Pressure plate – when the plate is pressed it will open the lid for couple of seconds, allowing the person to throw waste

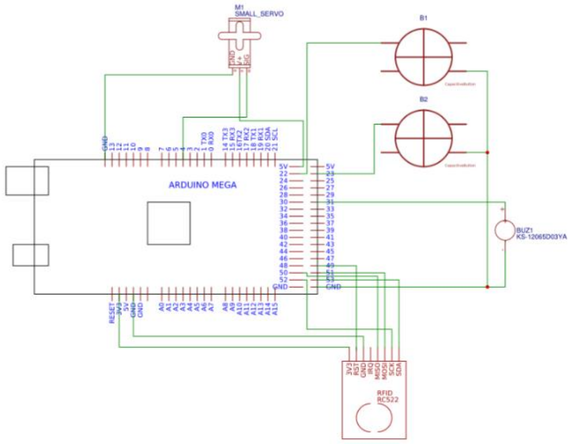


Fig. 2. Prototype opening system

C. *Fire detection system*

Being witnesses of many accidents of waste bins that are set on fire we put a flame sensor to detect ones. The data from this sensor is significant, hence it helps in reducing fires from spreading. This is done by sending information to the nearest firefighting unit about the location and the time where this fire has occurred.

D. *Volume measuring system*

For the volume measuring system we use ultra-sonic sensors attached to one servo motor.

1) *Rimeann sum*

We named our model after the famous German mathematician Georg Bernhard Riemann [6] who made many contributions to number theory and differential geometry. We came across his method for approximating the area under a curve using a finite sum. The sum is calculated by dividing a region into shapes (rectangular, trapezoids or parabolas) that form a region that is similar to the one being measured. By summing their areas, we approximate the region that is under the curve. Figure 3 show the Trapezoid Rule, that is a version of the Riemann method that uses trapezoids. We represent the empty space between the lid and the surface of the waste as the region that we want to calculate. Therefore, in real life the x-axis shown in Figure 3 would be the lid of the bin and the red curve the surface from the waste.

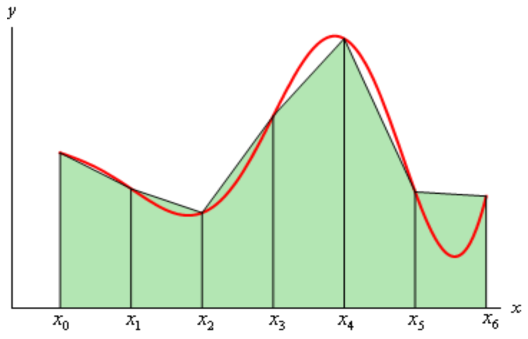


Fig. 3. Trapezoid Rule

2) *Formula*

The points on the x-axis that are shown on Figure 3 are the ultra-sonic sensors that we use to measure the distance. Equitation (5) is the an example of how by using the distances measured from N ultra-sonic sensors (the $f(x_k)$ values for $k = 1,2,\ldots N$) and the distances between the sensors (the Δx_k values for $k = 1,2,\ldots,N$) we create trapezoids. By adding these trapezoids, we have calculated the approximated area from that region.

$$\sum_{k=1}^{N} \frac{f(x_{k-1}) + f(x_k)}{2} \Delta x_k = \frac{\Delta x}{2} (f(x_0) + 2f(x_1) + \cdots + f(x_n)) \qquad (7)$$

By placing these ultra-sonic sensors on a stick, attached to a servo motor, the approximation of the entire region has a new dimension. The servo rotates clockwise in different angles, allowing the sensors to calculate the regions in every rotation.

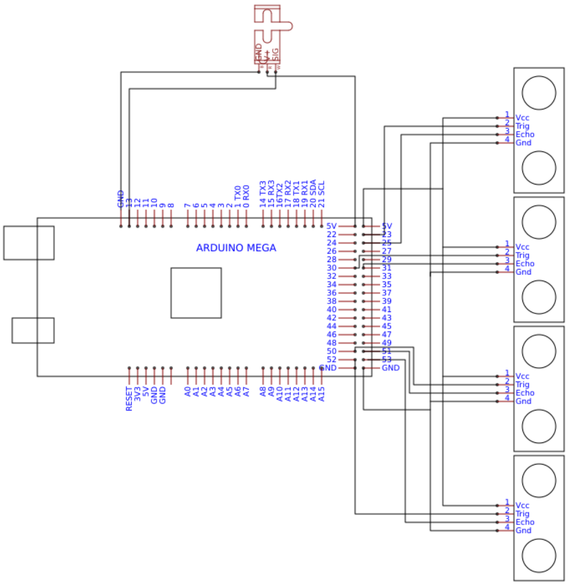


Fig.4. Prototype schematic for the measuring system

Figure 4 is a picture of how we resolved this issue. This way the system is covering more ground which leads to a better approximation. All the approximated areas are added together. Finally, in order to get the volume of this measuring we multiply this summed approximated area by rπ, where r is the radius of the lid (if the shape of the bin is cylinder which is our case study).

E. *Data transfer*

Sensor nodes are simple devices that can measure the empty space in trash bins using ultra-sonic sensors, and later transmit the data to the backend. Wireless communication is one of the key aspects of the design of the sensor node, and the overall topology of the system. Figure 5 represents the different technologies that are developed in the 21st century [8]. Some are likely to have high bandwidth (Wi-Fi), long range (GSM), high data rate (Cellular), low power (Bluetooth low energy (BLE)), or mesh-network capabilities (ZigBee). While all these technologies are considered to be mature, none of them are optimal for IoT projects. Generally, the problem is their power consumption. Higher data rate requires higher power.

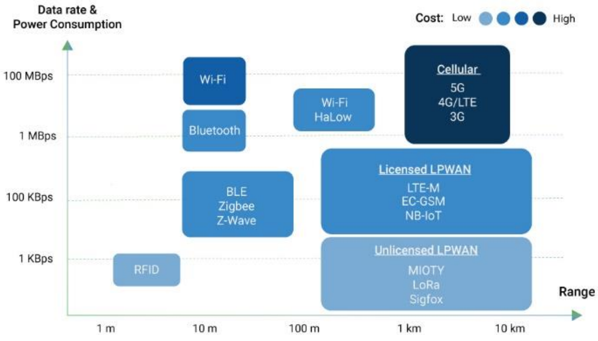


Fig. 5. Wireless technologies [8]

Whereas, the RGB model does not need to send pictures or videos it only needs report a percentage (the fill rate). That is why LoRa technology [9] is a perfect fit. This spread spectrum modulation technique derived from chirp spread spectrum (CSS) technology, which allows sending small messages on long ranges.

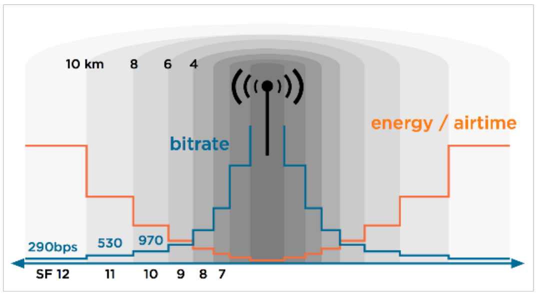


Fig. 6. LoRa spreading factor scalability [10]

The low data rate, uses less power, making this wireless platform the de facto technology for the IoT networks worldwide. Figure 6 shows the scalability of the spreading factor of this technology. As you can see smaller messages have higher airtime which means higher spreading factor [10].

F. *LoRawan / The Things Network*

LoRaWAN [11] is a media access control (MAC) protocol for wide area networks. It is designed to allow low-powered devices to communicate with internet-connected applications over long-range wireless connections. LoRaWAN can be mapped to the second and third layer of the OSI model. It is implemented on top of LoRa or FSK modulation in industrial, scientific and medical (ISM) radio bands. The LoRaWAN protocols are defined by the LoRa Alliance and formalized in the LoRaWAN specification.

LoRaWAN operates in unlicensed radio spectrum. This means that anyone can use the radio frequencies without having to pay million-dollar fees for transmission rights. It is similar to Wi-Fi, which uses the 2.4 GHz and 5 GHz ISM bands worldwide. The fact that LoRaWAN frequencies have longer range also comes with more restrictions that are often country-specific. In Europe, frequency band is in the 863-870 MHz frequency band. European frequency regulations impose specific duty-cycles on devices for each sub-band. These apply to each device that transmits on a certain frequency, so both gateways and devices have to respect these duty cycles.

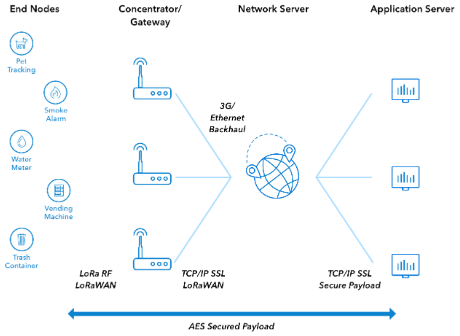


Fig.7. LoRaWAN architecture [12]

LoRaWAN architecture (Fig. 7) is also based around the use of nodes, gateways that – similar to Wi-Fi access points – pick up signals from the air and convert them, and a network server (an entire distributed infrastructure, in some cases) that effectively serves as data bridge to the application [12]. Figure 8 shows how the data transmitted by a node can be simultaneously picked up by multiple gateways, while encryption keys ensure that the network will accept the message and the application can process the decrypted data [13].

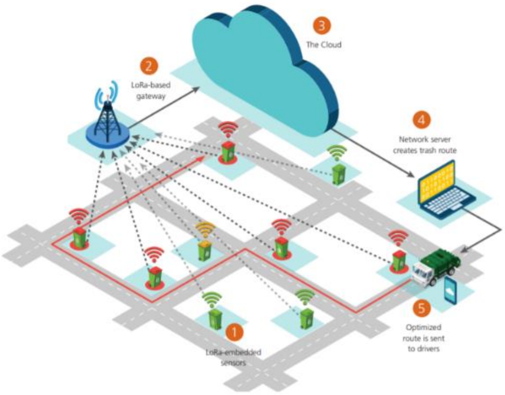


Fig. 8. LoRaWAN waste management system [13]

The Things Network [14] is a community-based initiative aimed at interlinking LoRaWAN gateways in order to create a large global network. The aim is to minimize the number of central components, while offering users the broadest possible range of options. Gateways can transmit data to multiple network servers, allowing for the creation of private networks and exchange data with TTN.

G. Backend

The backend consists of a cloud-based app that receives data from the nodes using MQTT (Message Queue Telemetry Transport) protocol [15]. MQTT is light weighted and requires limited network bandwidth, making it optimal for such short messages. The data is stored in a database which allows flexibility to test out what data might be useful to send a store without major changes. This also allows the implementation of the solution into any existing management systems.

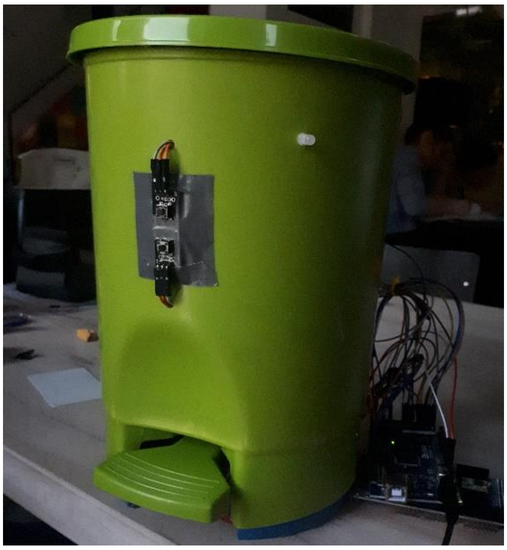


Fig. 9. Prototype

III. Conclusion

Making cities cleaner is provided with the idea of implementing a LoRaWAN waste management system of nodes. Waste management has become a reality rather than a problem. This case study based on placing sensors embedded with LoRa on existing waste bins, and through periodical reports of their fill rates, has shown that we can monitor fill level and reduce operational costs. As a result, this project through the IoT will improve the garbage management by relocating resources and by giving insight to the public about this global environmental problem.

References

[1] D. Constantino, Urban Smartness: Tools and Experiences

[2] R. Buchanan. Wicked probles in design thinking. Design issues, 8(2): pp 5-21, January 1992

[3] R. Giffinger. Smart cities. Ranking of European medium-sized cities: pp 13-18, 16 October 2007.

[4] D. Bongart, Analysis on the microcontrollers boom , avilbale at https://www.hannovermesse.de/en/news/the-microelectronics-boom-continues-82497.xhtml

[5] Arduino Mega 2560, avilable at, https://www.arduino.cc/en/Main/ArduinoBoardMegaADK?from=Main.ArduinoBoardADK

[6] R Dedekind, Biography of Riemann, in H Weber and R Dedekind (eds.), The Collected Works of Riemann (New York, 1953).

[7] Robert M. McLeod. The Generalized Riemann Integral. Definition of the Generalized Riemann integral, 1: pp 5-46, June 2014

[8] Different wireless technologies, avilable at https://iot-fpms.fandom.com/wiki/LPWAN_networks

[9] L. Vangelista, A. Zanella, and M. Zorzi. Long-Range IoT Technologies: The Dawn of LoRa™, pp 51–58. Springer International Publishing, Cham, 2015.

[10] Lora spreading factor, avilable at https://blog.surf.nl/en/lora-the-internet-of-things/

[11] Lorawan specification 1.0, lora alliance, 2015. avilable at https://lora-alliance.org/

[12] LoRaWAN architecture, avilable at The Things Network webpage, https://www.thethingsnetwork.org/docs/lorawan/

[13] Garbage Management Network by Semtech, avilable at https://www.semtech.com/uploads/technology/LoRa/app-briefs/

[14] N. Blenn, F.Kuipers. LoRaWAN in the Wild: The Things Network, 9 June 2017.

[15] MQTT description, avilable at https://iotfpms.fandom.com/wiki/MQTT

Sorting Algorithms and Their Complexities: A Case Study

Sara Temelkovska
School of Computer Science and Information Technology University American College Skopje
Macedonia
s_temelkovska@hotmail.com

Adrijan Božinovski
School of Computer Science and Information Technology University American College Skopje
Macedonia
bozinovski@uacs.edu.mk

Irena Stojmenovska
School of Computer Science and Information Technology University American College Skopje
Macedonia
irena.stojmenovska@uacs.edu.mk

Biljana Stojčevska
Faculty of Informatics University of Tourism and Management
Macedonia
b.stojcevska@utms.edu.mk

***Abstract*— A sorting algorithm is used for rearranging a given array or list of elements using comparison between the elements. There are different types of sorting algorithms but some of the most commonly used are: Selection Sort, Insertion Sort, Bubble Sort, Quick Sort and Merge Sort.**

In order to determine how efficient each of the sorting algorithms is, a desktop application was created, to sort arrays using the aforementioned algorithms. The array size and maximum element value can be chosen arbitrarily or randomly. One or all five of the sorting algorithms can be chosen to be tested, as well as the amount of times the tests should occur. The goal of the application is to count the sorting steps for each of the algorithms and to estimate which of the algorithms is the fastest and in which situations.

The application was tested and proven to work properly by giving valid results. We tested different situations: fixed array size and random maximum element value, random array size and fixed maximum element value, and both fixed and random array size and maximum element value. In the testing repetition, 30 tests for each array size up to 50 and random maximum element value was implemented. The conclusion/output is that the Bubble Sort algorithm is the slowest and the Quick Sort algorithm is the fastest.

Keywords— ***Complexity, Number of steps, Size, Sorting algorithms***

I. Introduction

In computer science and in mathematics, algorithms are used for solving classes of problems. Algorithms, data processing, calculations and automatic reasoning (which includes different views of reasoning) provide feasible solutions to various tasks.

An algorithm is an excellent calculating method, since it can be expressed within a well-defined formal language and a finite amount of time and space. Starting from one point, the instructions lead to a computation which, when executed, proceeds through a finite number of successive states which will provide an output at a final ending state.

Algorithms are used through all areas of IT (information technology). The concept of the algorithms exists for centuries, but to give a formal definition of algorithms remains a challenging problem until today. One of the most widely accepted (informal) definitions of an algorithm is that it is a finite set of instructions or rules that defines a sequence of operations for solving a particular computational problem for all problem instances for some problem set [1]. Later on, multiple variations (or improvements) of this definition have been given (e.g., [2], [3]). However, its essence remains the same.

II. Problem Statement and Solution

A. Theoretical Approach

1) Types of Algorithms

Algorithms with similar problem-solving approach are commonly grouped together. The algorithm types that we consider include: simple recursive algorithms, Backtracking algorithms, Divide and Conquer algorithms, Dynamic Programming algorithms, Greedy algorithms, Branch and Bound algorithms, Brute Force algorithms and Randomized algorithms.

The simple recursive algorithm solves the base cases directly, recurs with a simpler sub-problem and solves progressively more complex sub-problems, until the initial problem gets solved. The group of problems solvable by simple recursive algorithms contains: factorials, tree traversal, all permutations, quicksort, and Towers of Hanoi, among others.

The Backtracking algorithms are based on a depth-first recursive search. This means that the algorithm tests to see if the solution is found, and if it is, returns it. Examples of problems solvable using backtracking algorithms are the N-Queens Problem, Sum of Subset, Sudoku Puzzle, and Hamiltonian Cycle.

The Divide and Conquer algorithms consist of two parts. First, the problem is divided into smaller sub-problems of the same type and the sub-problems are solved recursively. Second, the solutions from the sub problems are combined into a solution to the original problem. An algorithm is called Divide and Conquer if it contains two or more recursive calls. Examples of problems solvable by this type of algorithms are: Binary search (which is a searching algorithm), quicksort (which is a sorting algorithm) and merge sort (which is a sorting algorithm too).

The Dynamic Programming algorithms remember the previous results from the past and use them to find new ones. This type of algorithm is most commonly used for optimization problems, for example: to find the “best” solution from multiple ones. The group of Dynamic Programming algorithms contains: Dijkstra’s algorithm for the shortest path problem, Fibonacci sequence, Towers of Hanoi puzzle, Checkerboard and Matrix chain multiplication.

A Greedy algorithm sometimes works well for optimization problems. This type of algorithm works in phases. At each of the phases, first the best solution is selected, without regards for future consequences. Examples of greedy algorithms are Kruskal’s algorithm and Prim’s algorithm for minimum spanning trees.

The Branch and Bound algorithms are used for optimization problems. When the algorithm progresses, a tree containing sub-problems is formed and the original problem is called the “root” problem.

A Brute Force algorithm tries all possibilities until a satisfactory solution is found. These algorithms can be optimizing. These algorithms can be made satisfying: they can stop when a good enough solution is found, provided that a metric for “good enough” is available and feasible.

The Randomized algorithms use a random number at least once during the computation to make a decision. An example of this type of algorithms is trying to factor a large prime by choosing a random numbers as possible divisors.

2) Algorithm Analysis

In theoretical analysis of algorithms it is common to estimate their complexity in the asymptotic sense, meaning to estimate the complexity for an arbitrarily large input. The big O notation, theta notation (Θ) and omega notation (Ω) are used for this.

The asymptotic estimates are usually used because of the different implementations of the same algorithm with different complexity. Though sometimes exact measures of complexity can be computed, they usually require certain assumptions, called models of computation [4].

The time complexity estimates depend of the fundamental step which we define. For the analysis, the time that is required to perform a fundamental step must be bounded by a constant. The space complexity estimates depend on the fundamental storage location which can be defined. The fundamental steps of that storage must support writing and reading functions, for which the time and/or space complexity can be calculated.

3) Sorting Algorithms

A sorting algorithm is used for rearranging a given array or a list of elements using comparison between the elements. The comparison operator is the one which decides the new order of the elements. There are different types of sorting algorithms but some of the most commonly used are the Selection Sort, the Insertion Sort, the Bubble Sort, the Quick Sort and the Merge Sort algorithms.

The Selection Sort algorithm sorts an array by repeatedly finding the minimum element (ascending order) from an unsorted part and putting that element at the beginning (Fig. 1). This algorithm has O (n^2) complexity, making it inefficient on large lists. It manages two subarrays from a given array: the already sorted subarray and the remaining subarray which is not sorted yet. With every iteration of this type of algorithm, the minimum element from the unsorted subarray is picked and moved to the sorted subarray.

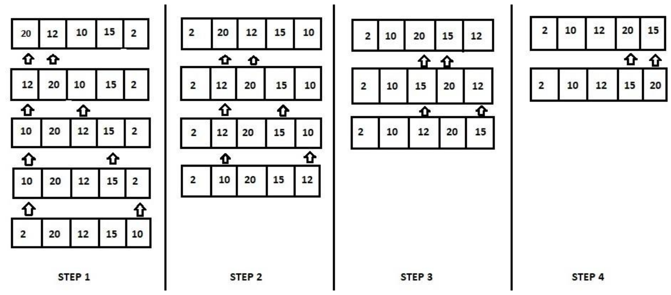
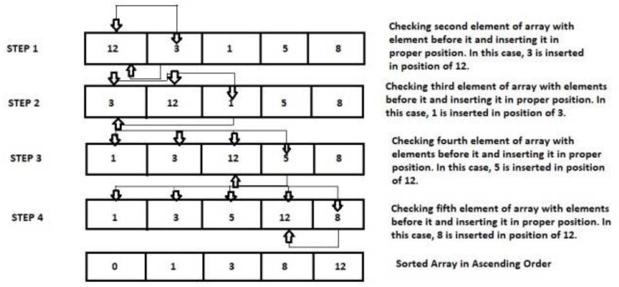


Fig. 1. Selection Sort Example

The Insertion Sort algorithm iterates consuming one input element by each repetition and the sorted output list is growing (Fig. 2). At each iteration, this algorithm removes one element from the input data, finds its location into the sorted list and inserts it there. This repeats until there are no input elements left. This algorithm also has O (n^2) time complexity.



Checking second element of array with element before it and inserting it in proper position. In this case, 3 is inserted in position of 12.

Checking third element of array with elements before it and inserting it in proper position. In this case, 1 is inserted in position of 3.

Checking fourth element of array with elements before it and inserting it in proper position. In this case, 5 is inserted in position of 12.

Checking fifth element of array with elements before it and inserting it in proper position. In this case, 8 is inserted in position of 12.

Sorted Array in Ascending Order

Fig. 2. Insertion Sort Example

The Bubble Sort algorithm is a simple sorting algorithm that is comparison-based in which each pair of adjacent elements is compared and the elements are swapped if they are not in order (Fig. 3). This algorithm's time complexity is O (n^2), so it is rarely used to sort large, unordered data sets.

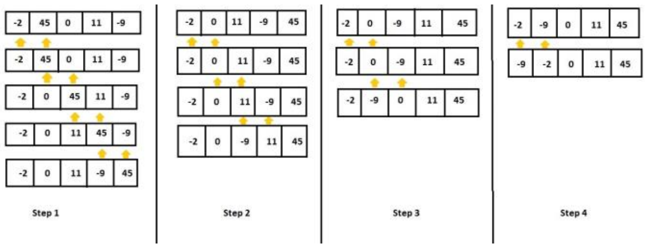


Fig. 3. Bubble Sort Example [5]

The Quick Sort algorithm is a Divide and Conquer algorithm. Its average time complexity is O (n · log n), and this is a popular sorting algorithm. It first selects a value which is called the pivot value. The main role of this value is to assist with splitting the array. The actual position where the pivot value belongs is the final sorted array also called the split point which will be used to divide the array for subsequent calls to the Quick Sort (shown on Fig. 4).

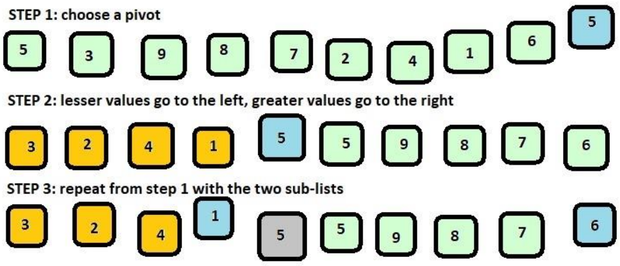
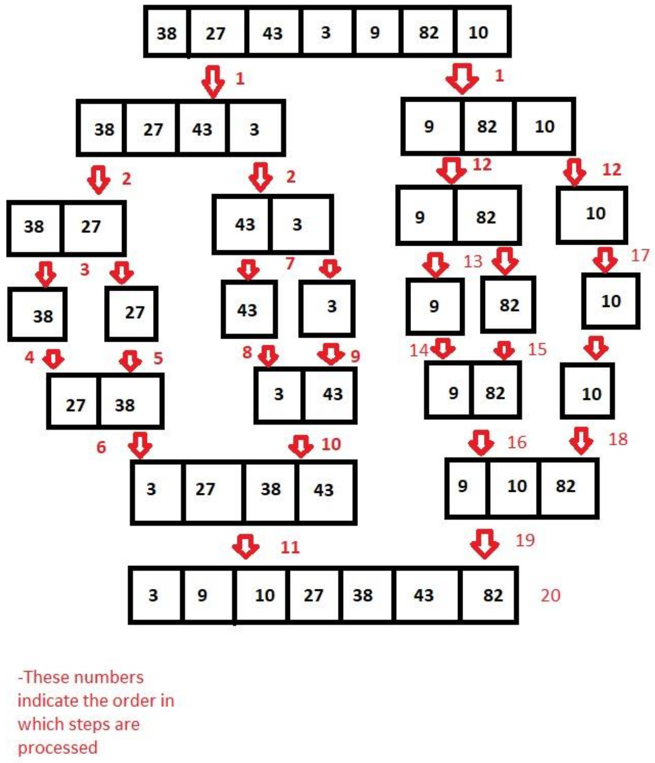

STEP 1: choose a pivot
5 3 9 8 7 2 4 1 6 5
STEP 2: lesser values go to the left, greater values go to the right
3 2 4 1 5 5 9 8 7 6
STEP 3: repeat from step 1 with the two sub-lists
3 2 4 1 5 5 9 8 7 6

Fig. 4. Quick Sort Example

The Merge Sort is comparison-based sorting algorithm which is also a divide and conquer algorithm. This is the first of the algorithms described here that scales well to very large lists, because its worst-case running time is O (n · log n). It first divides the array into the smallest unit (1 element), then compares each element with the adjacent array to sort and merge the two adjacent arrays. Finally all the elements are sorted and merged (Fig. 5).



-These numbers indicate the order in which steps are processed

Fig. 5. Merge Sort Example

B. Computer Application Description

For the development of this application, the C# language and the Microsoft Visual Studio development environment have been used. In order to see how each of the sorting algorithms performs, a desktop application was created (shown on Fig. 6).

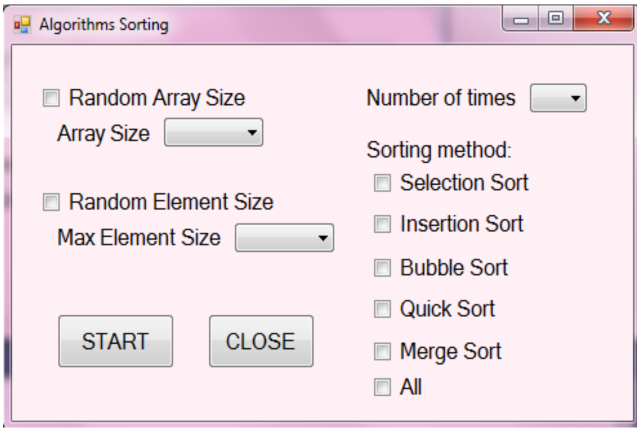


Fig. 6. Algorithms Sorting Application

The application has multiple options, for example one can choose the array and element size arbitrarily or they can be chosen randomly. The array size represents the number of elements in the array and the maximum element size represents the maximum value that can be assigned to any one of the elements of the array. There is also an option to choose between the sorting algorithms, i.e., one or a couple or all of the sorting algorithms can be (arbitrarily) chosen to be executed on the array of elements. The sorting can be repeated from 30 to 100 times, by generating new array values that have the same array size (unless it is specified to be random) and the same maximum element size (unless it is specified to be random). The goal of the application is to count the sorting steps for each of the algorithms and to see which of the specified algorithms is the fastest sorting and in which situations.

III. Results

For the proof of concept, our application was tested and proven to work properly by giving valid results. We tested the following situations: fixed array size and random maximum element size, random array size and fixed maximum element size, and both fixed and random array and maximum element size, whereas the array size was increased until (and including) 50.

The results are shown on Fig. 7. They represent an average of the times each array size has been accessed while being sorted, whereas the average has been taken out of 30 trials each array size was sorted using the specified sorting algorithms. At the beginning of every sorting trial each array was generated with a specified size, but random values of the elements, including the random maximum element value and each such array has been sorted using the sorting algorithms specified on Fig. 7. Thus, each point on the x-axis represents an average number of times 30 arrays with identical specified sizes but different elements have been sorted with each of the specified sorting algorithms. The amount of 30 sorting trials for each array size has been chosen because of the randomness of the values of each initial array, so sorting 30 arrays with the same array sizes but random-valued elements would provide a statistically significant average number of times the array has been accessed during each sort. The array is said to have been accessed every time an element of it has been used in any operation, including comparison and swapping.

The outcome of this proof of concept is that the Bubble Sort is the slowest and the Quick Sort is the fastest sorting algorithm.

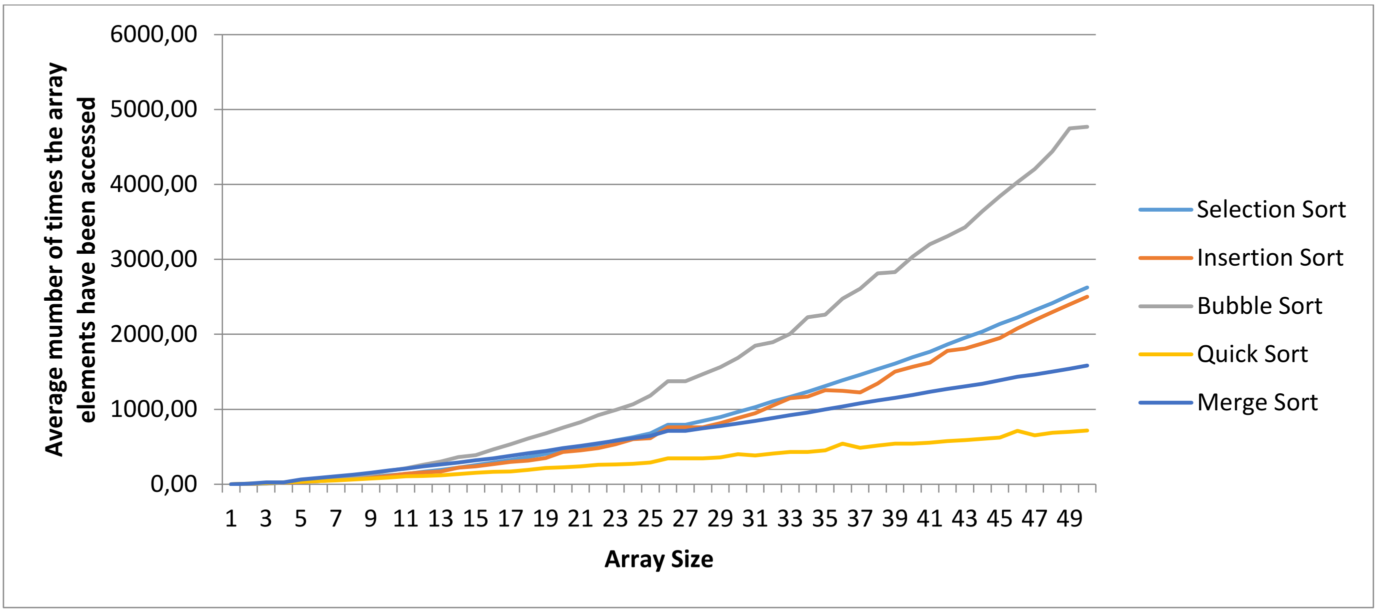


Fig. 7. Results from the Sorting Algorithms Application Proof of Concept Testing

IV. Conclusion

The goal of the development of the presented application is to create a starting point for future research that would hopefully produce interesting and applicable results. Our testing application was designed to generate random array sizes and maximum element values. For this proof of concept, the maximum array size that was used was 50, with random maximum element values. For each such array 30 iterations of sorting with Selection Sort, Insertion Sort, Bubble Sort, Quick Sort and Merge Sort were performed. The results showed that the Bubble Sort algorithm was the slowest in sorting such arrays and the Quick Sort was the fastest.

References

[1] D. E. Knuth, *The Art of Computer Programming: Volume 1: Fundamental Algorithms,* 3rd ed., Addison-Wesley, 1997.

[2] J. D. N. Dionisio, “Algorithms”, 2015 [Online] Available: https://dondi.lmu.build/share/intro/algorithms.pdf. Accessed on: Apr 6, 2019.

[3] Techopedia, “What is an Algorithm? – Definition from Techopedia”, 2018 [Online] Available: https://www.techopedia.com/definition/3739/algorithm. Accessed on: Apr 6, 2019.

[4] J. P. Gibson, “Complexity & Algorihtm Analysis”, 2012 [Online] Available: http://www-public.tem-tsp.eu/~gibson/Teaching/MAT7003/L9-Complexity&AlgorithmAnalysis.pdf. Accessed on: Feb 20, 2019.

The DevOps Toolbox: Be up to date with version-checker

Atanas Kostovski*, Panche Ribarski†
Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje
Email: *atanas.kostovski@students.finki.ukim.mk, †panche.ribarski@finki.ukim.mk

***Abstract*—There is a big problem with following updates for a big list of code repositories. The modern way of writing software and maintaining that software in production needs up-to-date versions of many software stacks and libraries. We propose a tool for the DevOps department (among others) which subscribes to a list of GitHub repositories and alerts to a Slack channel when a new version is available.**

***Index Terms*—DevOps, up-to-date, watch updates, GitHub, Docker, Kubernetes**

I. Introduction

The DevOps field of work is in the rise in last couple of years. Although not fully specified, this field has big range of responsibilities in the day-to-day activities in the IT segment. If there is one big thing that anyone agrees on in the DevOps community, that would be the process of automating things. This paper introduces the problem of following a big list of code repositories and watching for new releases in them. We propose a simple tool which works in stand-alone mode or running on some container orchestrator. The tool named version-checker watches for new releases on GitHub and alerts when a release happens with small info and direct link in a Slack channel. This allows automating the repetitive job of constant checks for new releases in a list of code repositories.

A. Related work

Our main accent is watching for new releases on GitHub repositories, therefore we first looked on GitHub for such a tool. We couldn't find a feature, but we found an issue [1] dating from 2015. Following this issues we found out that there is no such feature in GitHub, and further more, this feature is not enlisted in any public roadmaps for GitHub.

By doing additional search for this kind of tools, one of the most promising ones is [2]. This tool is very versatile because it supports a range of sources for watching releases. However, this tool uses cumbersome ini-like configuration and can't be used with an alerting system. We found a tool for watching GitHub repositories and alerting to Slack channel [3], but the configuration required that you need to pass a list of repositories when you run the container. This type of configuration didn't fit our requirements for the tool. The next tool we came to was [4] - unfortunately this tool was offered only as a front-end solution which allows you to register repositories and get email alerts, but the real back-end solution is not published by the author. Another tool which watches for new releases on GitHub and send email notification is [5]. The lack of notifications on Slack channels was the downside of this tool, but it's worthy noting that this tool is open-sourced completely.

Two of the proprietary tools that allow following updates are [6] and [7]. Sibbell [6] is shutting down on May 15th and CodeReleas [7] is a proprietary software which didn't fit in our requirements. We found few tools which were specific to other code repositories, such as [8] which is tied to watching npm packages.

B. Organization of the paper

The organization.

II. The big problem with following updates

Creating software nowadays is a tedious process because it's usual to use many other software stacks and libraries and build upon them. Further more, after finishing the software (if ever) the process of maintaining gets the scene. In a world of constant security breaches of the mentioned software stacks and libraries, you must follow updates and patches and implement them in the built software. In the DevOps field it's not just written software that has to follow updates. The platform on which the software resides (various servers, orchestration and other software stacks needed for the whole architecture) is also very dependent on following updates and implement those updates. For instance, a Kubernetes cluster for a simple web application has at least ten different software components which you should actively follow for updates: kubernetes, docker, traefik, flannel, kube-dns, postgres, solr, zookeeper, etcd and redis; and we actually can expand this list if the web application is a little more complex and requires additional components. In a microservice architecture the support stack is further more expanded with the needed roles such as load balancers, edge services, API gateway, service registry etc.

The number of components that need updates bring big responsibilities to the maintainers of the code and the servers to run it. This introduces almost daily checkups of the software repositories to see if there is a new version and what news it brings. The lack of such a tool in today's largest code repository, GitHub, brought us to the necessity of a tool which would follow a list of repositories in GitHub and alert when there is a new release. Searching for such a tool brought us to few possible tools, each of them either over complex or lacking some feature we needed. Therefore, we made the version-checker as a simple tool addressing the huge problems with following big number of code repositories and their releases.

III. The version-checker as a solution

The goal was to create a solution that would send notifications in a defined Slack channel, when a new release is published to the tracked Github repositories. This is done by using the public github api to check for new version, tracking the previous version with the etcd key-value store, and publishing notifications in a defined slack channel. The application is designed to be deployed on a Kubernetes cluster, but can be run with the provided docker-compose file too.

A. The Github API

Github is the most popular git platform for open source development, and is hosting many popular software repositories. The platform provides a public REST API for gathering information from all public projects. In this case, the only needed one is tracking new releases from the listed repositories. In Github as a platform, the term Release is a higher level concept built on top of the basic git feature, tags. Users can publish git tags as releases from the web interface, or have all tags listed by default if there is no published release.

Few details from the API documentation are quite important, in order to pick the right endpoint for tracking new releases. The API endpoint for listing releases returns only tags that are associated with a release from the interface, so if the repository has no such tags, returns empty array. It turns out, listing tags instead is more reliable, as it returns list of all tags, whether there are releases associated with them, so it works in any case. Worst case scenario is if the repository is set up in a way that not all tags are releases, but only tags associated with release manually, so it causes tracking of unnecessary information.

Listing tags is done with a GET request on the endpoint:

```
GET /repos/:owner/:repo/tags
```

As a response, the API provides a complete JSON list containing all tags.

B. Python to the help

Python script is used for solving all tasks in the project. After obtaining the list of repositories to be tracked, the latest tag of each repository is checked if it's already recorded in etcd. All data in etcd is stored in the /version-check directory, with keys in the repository/tag directory, and the latest recorded tag as a value of the key. When a newer version is found than the one stored in etcd, the tag stored is replaced with the new one and an alert is sent to Slack. If there is no record of the newly added repository, the latest tag found is handled as a new release.

There are two main ways of publishing messages to a Slack channel, by using the official client SDK provided, or the public HTTP api. The official client SDK is targeted for fully featured bots and chat applications, but the one for sending messages is similar with manually building the message and properties. The HTTP API is enough for this application, because it's only action to Slack is sending message, and building the message object is really similar in both implementations. This way, the message is built as a python dictionary, which is later serialized to JSON, and sent to the Slack hook url via POST request.

C. Try it with docker-compose

Docker-compose environment was created for testing purposes, but can be used for one-time running too. The default image set from docker hub, is built with the latest version of the script, and includes tracking of the default examples in repositories.txt. To change or add new repositories, new image should be built. It's recommended to track the versions manually, with tags. To build the image, run:

```
docker build -t owner/version-check:tag
```

The docker-compose environment starts up its own etcd, because the app relies on etcd for storing the latest versions found. To run this environment, there is an .env file that has to be edited. The default variables for the etcd connections are working for the local etcd started with the docker-compose yaml file, so only the slack hook and channel must be set. The slack hook is obtained by adding a new custom integration application to the slack workspace.

The variables needed for the setup are:

- `SLACK_HOOK=insert_slack_hook`
- `SLACK_CHANNEL=#change-to-your-slack`
- `ETCD_HOST=etcd1 # etcd hostname`
- `ETCD_PORT=2379 # Change to etcd port`

D. Deploy in Kubernetes

The application is a one time run only, and returns a success code on completion. The docker compose file can be run for checking versions once, or schedule the startup with tools like cron, but the project is designed to be run as a Kubernetes cron job. The provided deployment file is set to be run once a day, and restarts if the script returns non-zero code.

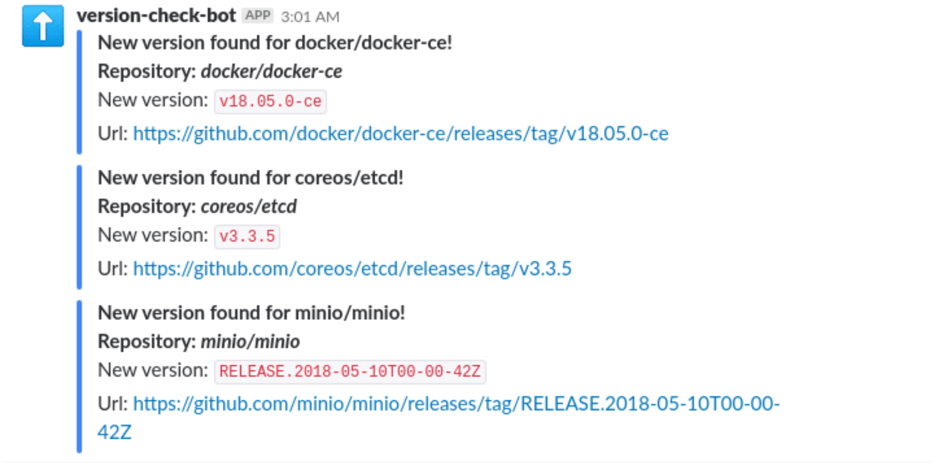
To run the application, the same environment variables like with docker-compose are needed to be updated for the Slack connection, plus the values for the ETCD connection. The schedule job is started in a namespace called version-check, which can be created with the command:

```
kubectl create namespace version-check
```

The image in kubernetes can be set to the latest tag. With that kind of configuration, when it's rebuilt and updated, because of the Always image pull policy the next run should include the new image with the updated repositories to track. It can also be manually updated for better tracking of the running version.

Deploy the schedule job with:

```
kubectl create -f ./version-check-cronjob.yaml
```



version-check-bot APP 3:01 AM

New version found for docker/docker-ce!
Repository: ***docker/docker-ce***
New version: `v18.05.0-ce`
Url: https://github.com/docker/docker-ce/releases/tag/v18.05.0-ce

New version found for coreos/etcd!
Repository: ***coreos/etcd***
New version: `v3.3.5`
Url: https://github.com/coreos/etcd/releases/tag/v3.3.5

New version found for minio/minio!
Repository: ***minio/minio***
New version: `RELEASE.2018-05-10T00-00-42Z`
Url: https://github.com/minio/minio/releases/tag/RELEASE.2018-05-10T00-00-42Z

Fig. 1. A Slack message from version-check alerting for new releases.

IV. Conclusion

Following updates to software stacks and libraries used in your software and on your systems is very important. The new concept of releasing often and the many security breaches and security patches mandates that the developers and DevOps should constantly look for new versions and releases of the used software blocks. The tool version-checker helps in the repetitive job of watching GitHub repositories and informs you on a Slack channel when there is a new release. This helps with aligning with the newest updates and keeping the built software and systems always up-to-date.

This work was partially financed by the Faculty of Computer Science and Engineering at the ”Ss. Cyril and Methodius University” within the project SEV.

References

[1] “Ability to watch a project but only for releases - github issue 410,” https://github.com/isaacs/github/issues/410, accessed: 2018-05-08.
[2] “New version checker for software releases,” https://github.com/Richard87/releaser, accessed: 2018-05-08.
[3] “github-releases-notifier - receive slack notifications for new releases of your favorite software on github,” https://github.com/justwatchcom/github-releases-notifier, accessed: 2018-05-08.
[4] “Releaser - a small angular4/firebase app that monitors github repositories for new releases,” https://github.com/Richard87/releaser, accessed: 2018-05-08.
[5] “Watch for releases on github and get free email notifications,” https://github.com/vfeskov/gitpunch, accessed: 2018-05-08.
[6] “Watch changes with sibbell,” https://about.sibbell.com/, accessed: 2018-05-08.
[7] “Coderelease - never miss a new release again,” https://coderelease.io/, accessed: 2018-05-08.
[8] “Update notifications for your cli app,” https://github.com/yeoman/update-notifier, accessed: 2018-05-08.

The Geographic Flow Of Music On Spotify

Lidija Jovanovska
Faculty of Computer Science
and Engineering
Skopje, Republic of North Macedonia
lidija.jovanovska@students.finki.ukim.mk

Igor Mishkovski
Faculty of Computer Science
and Engineering
Skopje, Republic of North Macedonia
igor.mishkovski@finki.ukim.mk

Miroslav Mirchev
Faculty of Computer Science
and Engineering
Skopje, Republic of North Macedonia
miroslav.mirchev@finki.ukim.mk

***Abstract*—As Daniel J. Levitin interestingly noted, No known human culture now or anytime in the recorded past lacked music. Therefore, the impetus behind this research paper is to model the interactions between countries in order to reveal music listening trends at a macro level. Subsequently, the framework for performing this analysis consists of techniques used in the multidisciplinary field known as Network Science. Throughout the past decade, the world has witnessed a gradual shift in the way music is listened to. In that respect, Spotify, an online music streaming service, has been the imperative giant with a user base of around 191 million. With the help of Spotify's Application Programming Interface (API), a dataset was compiled, which contains the Weekly Top 40 streamed songs across 50 countries, in the year 2017. Through research, the team explored whether, and to which extent, do language, nationality and geographic distance influence the way global communities are formed. Furthermore, the project aimed to prove that there is a clear direction of leadership flow in the network. Until now, the acquired information supports the hypotheses that some countries do indeed follow the trends beset by others and that language and nationality play an essential role in the development of communities.**

***Index Terms*—music, network science, clustering, leadership**

I. Introduction

Even though music is deeply rooted into mankind's history, people began recording it in the mid-1850's. After years of improving and innovating on recording technologies, mankind reached the Internet era, which increased the ease of access per user, as well as enrich the music world with a wide variety of artists and genres. As the efficiency and availability of physical music recordings waned in recent years, a substantial part of the population decided to cross over to music streaming services. While physical and download revenue continue dropping worldwide, digital and streaming services are experiencing the exact opposite. Streaming revenue increased by 230% between 2013 and 2017, marking it as a period of consistent growth.

Spotify is undoubtedly the most popular platform in the music streaming business, harboring a user-base of around 191 million. The platform also provides a powerful Application Programming Interface (API), through which Music Information Retrieval (MIR) researchers can gain access to audio features, artist information, as well as daily and weekly global streaming charts. Due to Spotify's dominant position in the business, the data analyzed in this project was acquired through their API because it provides current relevant information regarding each country's streaming preferences.

Lao and Nguyen analyzed data from the 'Billboard Hot 100' charts spanning from 1958 to 2015. They found that the switch to digital technology significantly lowers the cost of publishing a single hit, enabling already popular artists to increase their fame and consequently homogenize the top charts. As Ferreira and Walfdogel found in their 2010 research on the global music trade, from 2001 to 2007, 31 artists have appeared simultaneously on at least 18 countries' charts in at least one year [1]. However, digital songs are more likely to fall off the chart in the first week compared to CD songs, which indicates a highly volatile nature, resulting in heterogeneous charts on a regular level [2].

Our analysis of the geographic distribution of musical preferences is structured as follows: We begin by describing the data, a world-wide log of streaming habits recorded by Spotify, as well as various preprocessing steps in section III. In the following step, in section IV, we describe how we constructed the network and investigate whether and to which extent communities are formed based on language and geographic distance.

In section V we describe how we adapted and adjusted a methodology previously used to find leadership in pigeon flocks to detect leader-follower relationships between countries. The methodology involves examining every dyad between the countries in the dataset and testing whether the time-lagged correlation is significantly larger in one direction than the other.

II. Related Work

The field of Network Science provides a framework for modeling interactions between entities so as to reveal properties at a macro level, which may not be noticeable or visible at the individual level. Techniques from this area of study have been successfully implemented to many other fields including Music [3].

Nagy et al. provided a powerful methodology for detecting leader-follower pairs, which was previously applied in the search for the leadership hierarchy present in pigeon flocks [4]. This was further adapted by Lee and Cunningham in their research about the global flow of music on Last.fm, which served as the incentive for this paper [5].

Shafiq, Ilyas, Liu and Radha developed a model for identifying specific types of leaders, followers and neutrals. The model was applied on data from Facebook and was ultimately able to capture the characteristic differences of the user categories [6].

III. Data

A. Preprocessing

Spotify keeps record of the daily Top 200 Charts for every available country on a dedicated site. The data can be scraped systematically by downloading the .CSV files separately for each country. The scope of this analysis will be the entire 2017 year from January 1st, 2017 to January 1st, 2018, inclusive.

Because not all Spotify users are consistently daily active, a single day's chart can be thought of as a sample of listening preferences among users. In countries that have relatively few users, the variance associated with this sample becomes large, indicating high volatility. This noise can be reduced by aggregating a matrix associated with seven consecutive days together. By doing that, it effectively increases the sample size for each entry in the country-song matrix.

For some countries, there was not sufficient data to compile a Weekly Top 200 list for every week in the entire year. Since record charts have traditionally consisted of a total of 40 songs, it was decided that a Weekly Top 40 Chart will represent a country's weekly streaming preferences accurately and avoid making a significant reduction to the country list.

B. Missing Data

Despite downsizing the dataset, there was no sufficient data to compile the Weekly Top 40 lists for several of the countries that newly adopted the service, including Lithuania, Luxembourg and Estonia. Consequently, those countries were removed from the dataset, resulting in a final list consisting of 50 countries. By doing this, the size of the dataset reduced significantly, while the number of unique songs dropped from 21,747 to 1551, indicating that the Top 200 Charts are significantly more heterogeneous than the Top 40 Charts.

C. Creating Streaming Matrices

In order to create a suitable representation for a country's streaming history, we aggregate the data in so called 'streaming matrices'. In that manner, we have a matrix for each week. In this matrix every country is a row vector with 1551 elements, and each column represents a song. Each song in the vector defines a dimension in Euclidean space and the frequency of each song corresponds to the value in that dimension. Since not all countries have the same set of songs appearing on their Top 40 chart, there is a large number of zero-valued elements, resulting in sparse streaming matrices. Therefore, a non-zero entry in the matrix at position i, j is a positive integer, indicating the total number of times the users from country i have streamed the song j in that particular week.

IV. Clustering: Geographical clusters are strong

A. Creating the network

With the purpose of comparing the streaming vectors of each pair of countries, we must choose a similarity measure.

Not all similarity measures yield the same results in a certain scenario. Therefore, it is essential to choose the metric that will describe the data properly. In this project, we considered the use of two measures: cosine similarity and jaccard similarity.

By using jaccard similarity, we risk losing information regarding the songs' streaming frequencies that may prove to be useful when deciding how to generate the network. During testing, cosine similarity yielded more plausible results, as expected, and was subsequently used when measuring the similarities of each pair of countries' streaming preferences.

Determining the threshold. In our graph, each node represents a country, while each link between a pair of nodes indicates that those countries have a similarity index above a certain threshold. The determination of the threshold here is heuristic, i.e. it is not validated by human subjects, nor estimated by a predictive model. As a result of extensive testing, a threshold of 0.3 was settled on.

Since 2014, the average number of artists each Spotify user streams per week has increased by 37 percent. So far, in 2017, it rose from just under 30 to about 41 different artists per week. This fact might lead us to believe that the rise of streaming services has increased heterogeneity across musical charts. To test this claim we attempt to find whether clusters are formed based on language and geographic distance.

To construct the dendrogram shown in figure 1, we performed average linkage clustering (an agglomerative clustering algorithm) on the adjacency matrix A of the cities, a square matrix where each entry A(i,j) is the cosine similarity between country i and country j. Instead of constructing the dendrogram based on just a single streaming matrix, we summed together the similarity matrices spanning from January, 2017 to December, 2017.

B. Discussion

Starting from the lowest level structure of the tree, we can already see clusters of pairs of countries which are geographically close to each other. Such examples include: The United Kingdom and Ireland, New Zealand and Australia, Germany and Austria, Slovakia and the Czech Republic, Bolivia and Ecuador, Chile and Peru etc. There are some exceptions to this claim, notably Latvia, which seems to be more similar to New Zealand and Australia, than to other European countries. For example, it is strange to see The United States being closer to Iceland, rather than to The United Kingdom, despite the fact that Iceland is the closest European country to The United States.

At an intermediate level, we can observe that most clusters are formed based upon language. There are two clusters

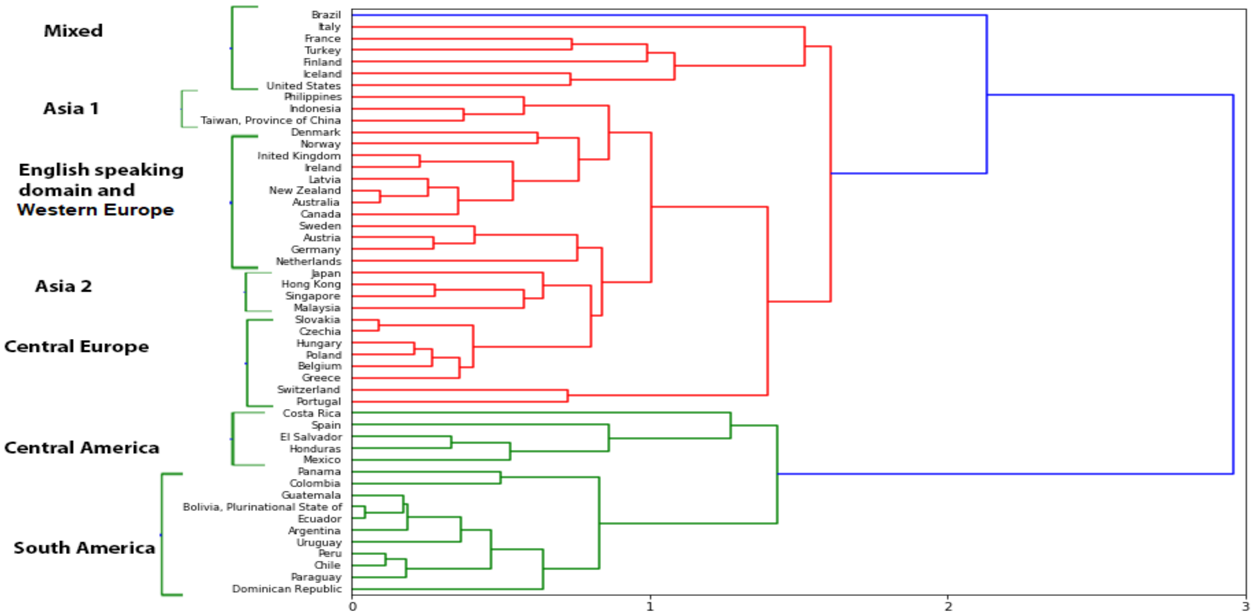


Fig. 1. Hierarchical clustering based on average linkage clustering of cosine similarities of countries in the normalized streaming matrices.

of Spanish speaking countries; one consisting of Spain and countries from Central America, and another, consisting of countries from South America. An English dominated cluster emerges, with Latvia being the odd one out. The Eastern Asian Countries form two clusters which despite their geographical adjacency, are not close at all regarding their similarity scores. The rest of the clusters include Slavic speaking countries, along with Greece and Belgium, and three clusters consisting Western and Northern European countries. It is surprising that Switzerland and Portugal form a cluster even though there is no overlap in the sets of nationalities that inhabit those countries, nor is there a common language. The multinational character of Switzerland would be expected to be a key factor in high similarities between Germany, France and Italy. The strangest cluster at this level is definitely the one constituted of countries such as: Iceland, United States, Turkey, Finland, France and Italy.

At the next level of the hierarchy, there are two notable clusters. Here, language seems to play a key role in the formation of these clusters. The first cluster consists exclusively of Spanish speaking countries, while the other includes every country left in the dataset, except Brazil. We would expect Brazil to be relatively closer to the first cluster, due to the geographical proximity and the fact that Portuguese and Spanish are similar languages. However, to our surprise, Brazil's community joins the second cluster on a relatively higher level. We speculate that Brazil does not belong to a particular community because it's the largest country in South America and hence it might have its own distinct idiosyncratic preferences.

V. Methodology

A. *Leaders and Followers*

To detect leader-follower pairs, the methodology of Lee and Cunningham was implemented, which is based on finding lagged correlations, and was also previously applied to finding leadership in global music listening preferences. When examining the relationship between a pair of countries, we are interested whether there is a directed link from one country to the other, or whether it is of neutral nature, where neither leads the other.

We begin by calculating the velocities for each country in the dataset. A velocity $v_{country}(t, t+1)$ represents the change that takes place in the listening habits of a country from one week t into the next $t+1$. That leaves us with a sequence of velocities for each country.

To measure whether a country i follows a certain country j, we measure the cosine similarity of each of the country i's velocities with the velocities of the country j from one week earlier. The average of these lagged similarities are referred to as the correlation of the first country's velocities with the second country's lagged velocities, where the lag size is one week. We call this measure C.

B. Deciding Which Edges To Accept

It could be the case for a dyad i, j that after testing whether a correlation is strong enough to be accepted, i appears to follow j and j appears to follow i. While it would be easier to choose the direction with the higher correlation, that option could mean that neither country is leading another, and instead, they are moving together.

To make sure there is a clear direction to the leader-follower relationship, we perform a t-test to make sure that the two correlations (which are means of similarities) are not equal; here we use a two-sided, paired t-test. Since our sample size is relatively small, we perform a Levene test to check whether the variances of the correlations are equal. If the test is positive, i.e. the null hypothesis is accepted, we proceed with the independent t-test. Otherwise, we use the t-test for related samples. If one correlation is larger, then we accept the leader-follower pair associated with that correlation as a directed edge; otherwise, it is concluded that no leader-follower relationship exists.

VI. Results

After adapting the methodology to the subject at hand, we find all leader-follower relationships in the network and then assign edges weighted by the lagged correlation. The graph which can be seen in figure 2 is a DAG (directed acyclic graph). The size of the nodes indicate their PageRank and the color represents the community to which they belong. In the past, it has been argued that a system with a strong leadership hierarchy ought to be nearly acyclic, so the lack of cycles in the output network is a clear validation of that theory [7].

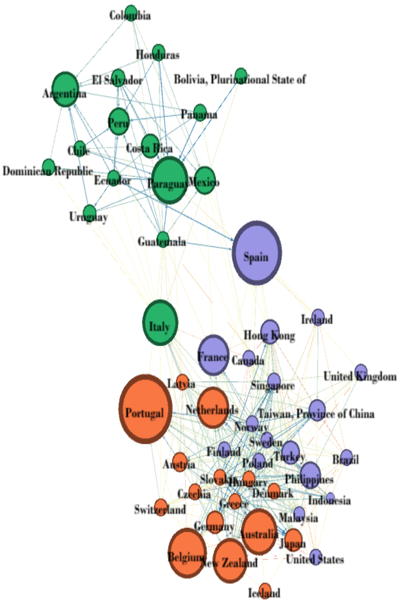


Fig. 2. The Geographic Flow of Music on Spotify

There are many centrality measures that could be used as criteria for deciding which countries are musical trendsetters and which are laggards. PageRank was used because it is an algorithm designed to rank importance of nodes on weighted, directed networks on which a dynamic process takes place [8].

As we can see in table I, Spain and Portugal are the most important nodes in the network due to the fact that they both serve as bridges between the South American and the Eurasian communities. This indicates that language and geography play a key role in streaming preferences, since both European countries use Latin languages. Both communities have no outgoing links and yet they are followed by some of the more active countries including The United Kingdom, Sweden, Norway etc. It is strange to see Australia and New Zealand's prominent rank, while we would expect The United States, United Kingdom and Japan to fare much better.

TABLE I
Statistics for the most important nodes in the network

Country	PageRank
Portugal	0.068441
Spain	0.062991
Belgium	0.047724
Paraguay	0.045235
Australia	0.043667

VII. Conclusion and Future Work

This exploratory data analysis aimed to find out whether countries form streaming communities based on language and geographic distance. There is evidence to support this claim, although there are a few exceptions. Furthermore, two aspects of the methodology give credibility to the results: that each leader-follower relationship underwent a t-test, and that when all of the leader-follower relationships were put together into a graph, they formed a DAG, indicating a direction of flow in a strict sense.

Future plans include obtaining a larger dataset and implementing the methodology on sets of songs belonging to various genres. By doing so, we can shed light on trends, hidden, due to the multi-dimensional aspect that genre brings to the data.

References

[1] F. Ferreira and J. Waldfogel, “Pop internationalism: has half a century of world music trade displaced local culture?,” *The Economic Journal*, vol. 123, no. 569, pp. 634–664, 2013.

[2] J. Lao and K. H. Nguyen, “One-hit wonder or superstardom? the role of technology format on billboards hot 100 performance,” 2016.

[3] C. A. Perrone and C. Dunn, “Brazilian popular music and globalization,” *Journal of Popular Music Studies*, vol. 14, no. 2, pp. 163–165, 2002.

[4] M. Nagy, Z. Akos, D. Biro, and T. Vicsek, “Hierarchical group dynamics in pigeon flocks,” *Nature*, vol. 464, no. 7290, p. 890, 2010.

[5] C. Lee and P. Cunningham, “The geographic flow of music,” in *2012 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, pp. 691–695, IEEE, 2012.

[6] M. Z. Shafiq, M. U. Ilyas, A. X. Liu, and H. Radha, “Identifying leaders and followers in online social networks,” *IEEE Journal on Selected Areas in Communications*, vol. 31, no. 9, pp. 618–628, 2013.

[7] E. Mones, L. Vicsek, and T. Vicsek, “Hierarchy measure for complex networks,” *PloS one*, vol. 7, no. 3, p. e33799, 2012.

[8] L. Page, S. Brin, R. Motwani, and T. Winograd, “The pagerank citation ranking: Bringing order to the web.,” tech. rep., Stanford InfoLab, 1999.

Vehicle Detection with HOG and Linear SVM

Nikola Tomikj
Ss. Cyril and Methodius University
Faculty of Computer Science and Engineering
1000 Skopje, North Macedonia
tomikj.nikola@students.finki.ukim.mk

Andrea Kulakov
Ss. Cyril and Methodius University
Faculty of Computer Science and Engineering
1000 Skopje, North Macedonia
andrea.kulakov@finki.ukim.mk

***Abstract*—In this paper, we present a vehicle detection system by employing Histogram of Oriented Gradients (HOG) for feature extraction and linear SVM for classification. We study the influence of the color space on the performance of the detector, concluding that decorrelated and perceptual color spaces give the best results. An in-depth analysis is carried out on the effects of the HOG and SVM parameters, the threshold for the distance between features and the SVM classifying plane, and the non-maximum suppression (NMS) threshold on the performance of the detector, and we propose values that illustrate good performance for vehicle detection on images. We also discuss the issues of the approach and the reasons for its mediocre performance on videos. Finally, we address these issues by presenting ideas that can be considered for improving the system.**

***Index Terms*—computer vision, machine learning, HOG, SVM, color space, vehicle detection, autonomous vehicles**

I. Introduction

In the past few years, autonomous driving has been gaining a lot of interest and it is expected to be the next big thing in the automotive industry. One of the main challenges in the development of the intelligence that powers the autonomous vehicles is its ability to detect obstacles like pedestrians, other vehicles and objects on the road. This ability provides the safety required to make autonomous vehicles mainstream. Different techniques for preceding vehicle detection have been developed throughout the literature, from traditional computer vision techniques to deep learning ones.

Dalal and Triggs [1] describe the HOG method in their breakthrough paper. They use the method for human detection. They performed thorough experiments with the parameters of the method. The values for the parameters that they found to be optimal for human detection are akin to the values that we found to be optimal for vehicle detection.

Creusen, Wijnhoven, Herbschleb, *et al.* [2] experimented with different color spaces and concluded that the choice of a color space significantly influences the performance of the HOG detector, and that the optimal color space choice depends on the type of object that the detector is trying to detect. They showed that for decorrelated color spaces like HSV, the performance when the H-channel is used as a single channel detector is almost identical to the performance in the HSV space. They concluded that this indicates that saturation and intensity information is largely irrelevant, and that color is the dominant feature. They also concluded that HSV and RGB are less suitable for traffic sign detection and detection of objects that have large color variation in general. They found out that the LAB and YCrCb color spaces provide the best performance, and that is probably due to the availability of two dedicated color channels.

Mao, Xie, Huang, *et al.* [3] took very similar approach to the one described in this paper, using HOG and linear SVM as well, and they developed effective and robust preceding vehicle detection system that can achieve high reliability target detection and low false positive rate.

An interesting and very useful approach was proposed by Arróspide, Salgado, and Camplani [4]. They overcame the computational limitations of the standard HOG, which yields excellent performance but can hardly be used in a real-time environment. They developed alternative HOG descriptors which are designed to be cost effective by making use of the previous knowledge on vehicle appearance.

Lately, deep learning approaches have been gaining popularity and several authors have experimented and produced state-of-the-art results. Fast R-CNNs are used extensively for object detection as proposed by Girshick [5]. Another important mention is YOLO introduced by Redmon, Divvala, Girshick, *et al.* [6].

In this paper we develop a pipeline for detecting preceding vehicles using HOG and linear SVM. Our aim is to study the HOG and linear SVM applicability and potential for vehicle detection. We experiment with the HOG and SVM parameters, the threshold for the distance between features and the SVM classifying plane, and the NMS threshold, to examine their influence on the performance of the system and to reveal values that provide good performance.

II. Methodology

We implemented a vehicle detection pipeline in Python that detects vehicles in images and videos recorded with a dashcam, using HOG and Linear SVM. The pipeline is developed in Python 2.7 [7] and OpenCV 3.4.3 [8].

The details of each component in the pipeline are presented as follow:

A. Data Preprocessing

The labeled data come from a combination of the GTI vehicle image database [9] and KITTI vision benchmark suite [10]. The data are png color images with dimensions 64×64, which is convenient for computing the HOG descriptors of the images. The labeled data contain 8792 vehicle images and 8968 non-vehicle images. The non-vehicle data also contain some images extracted from a real dashcam video with hard negative mining to reduce the number of false positives.

Because all training images have the same dimensions, their HOG feature vectors have the same length and can be used to train an SVM. So, preprocessing is not necessary, however converting the images to certain color spaces could potentially increase the performance of the detector.

B. Feature Extraction

After the preprocessing stage, feature extraction is performed by computing the HOG descriptors of every preprocessed image from the labeled dataset. These descriptors are used to train and test a linear SVM.

C. Training a Linear SVM

This is 2-class classification problem, so the viable SVM type options that OpenCV offers are C_SVC and NU_SVC types. As they are very similar, we decided on using NU_SVC.

We were bound to use linear SVM because only the primal form of a linear SVM can be passed as an argument to the HOGDescriptor struct setSVMDetector function. This function allows the usage of the performant detectMultiScale function which this pipeline is trying to make use of.

D. Vehicle Detection

The last step is to perform vehicle detection on real dashcam data. For this purpose, we used the detectMultiScale function to detect vehicles in images and videos. This function performs a sliding windows search using windows with different sizes, so that vehicles with arbitrary dimensions can be detected in the input image or video.

III. Results

A. Color Space

To determine the color space that provides the best performance, experiments using various color spaces were performed, using constant values for the HOG and SVM parameters.

Table I gives comparison of the performance for 8 different color spaces based on the error as a percent of misclassified images from the test set when the images are in certain color space. The values of the HOG and SVM parameters used for the experiments are given in table II. These are some common default values for these parameters.

Table I
Color Space Performance

Color Space	Error
BGR	2.5732%
GRAY	3.5867%
LAB	3.3531%
LUV	3.2573%
HLS	5.7742%
HSV	6.0923%
YCrCb	3.2404%
YUV	3.1757%

Table II
Parameters Testing Values

Parameter	Value
Detection window size	64×64
Block size	16×16
Block stride	8×8
Cell size	8×8
Bins	9
Discrete derivative mask size	1
Gaussian smoothing window parameter	-1 (means no smoothing)
Block normalization type	L2-Hys
L2-Hys threshold	0.2
Gamma correction	False
Signed gradient	False
Nu	0.09

Oddly enough, BGR gives the best performance, or the least percentage of misclassified samples. This was unexpected because the BGR color space has strongly correlated channels and is non-perceptual. But, although it achieves the best performance during the testing phase, it performs poorly when detection is performed on real images and videos. We think that the reason is that real images and videos have a lot of background color variation, so using BGR which has strongly correlated channels will cause many misdetections. That is the case because HOG calculates separate gradients for each color channel and the one with the largest norm is taken as the pixel's gradient vector. LAB, LUV, YCrCb and YUV all achieve similar performance, and follow after BGR. Any of these 4 color spaces is a viable option because the difference between their performance is very small. We decided to use the YUV color space because it has the smallest error percentage in table I and it works well with our test data.

B. HOG Parameters

HOG is the key component in the pipeline, so the choice of the values for the HOG parameters is a very important one. We experimented with different values for these parameters and found out that the ones suggested by Dalal and Triggs [1], shown in table II are performing reasonably well. The specific effects that the values of these parameters have on the performance i.e. the linear SVM testing error, are the following:

- Because the dataset images are 64×64 pixels, it is reasonable to use a 64×64 detection window size. Decreasing or increasing this size will decrease the performance because the images will contain less information, or it will make the descriptor more sensitive to noise, respectively.
- Block size, block stride and cell size depend on each other's value and influence the performance together, so they must be tested together. Table III shows the results of the experiments and gives insight about the effects of these three parameters on the performance. The interpretation of these results is that overlap of $3/4$ slightly improves the performance, however it drastically

Table III
HOG Parameters Performance

Block Size	Block Stride	Cell Size	Error
8×8	4×4	4×4	3.2751%
8×8	8×8	8×8	6.5991%
16×16	4×4	4×4	2.3341%
16×16	8×8	8×8	2.6858%
16×16	16×16	16×16	7.7984%
32×32	8×8	8×8	2.3649%
32×32	16×16	16×16	2.7703%
32×32	32×32	32×32	16.4527%

increases the time required for extracting the HOG features and training the SVM which makes it not worth it for such a small performance increase. $1/2$ overlap is enough.

- Discrete derivative masks of size 1 without Gaussian smoothing work best. Using larger masks decreases performance and smoothing decreases performance significantly. However, larger masks and Gaussian smoothing can be useful on real videos.
- Using more than 9 bins with unsigned gradients will not improve the performance significantly. Decreasing the number of bins decreases the performance. So, it is appropriate to use 9 bins for vehicle detection.
- The only block normalization type currently available in OpenCV is L2-Hys. The optimal L2-Hys threshold is 0.2, increasing or decreasing this threshold decreases the performance.
- Using gamma correction increases the error for around 0.5%, however it can be useful on real videos.
- Dalal and Triggs [1] used unsigned gradients for human detection in their paper, but they suggested that for some other tasks like vehicle detection sign information helps substantially. However, we found out that this is not the case. Signed gradients seem to cause overfitting of the SVM and cause false negatives on real images and videos. Using signed gradients also increases the time required for extracting features and training the SVM, because the feature vectors become longer (assuming the number of bins is 18) and therefore, the pipeline is slower. So, we think that one should stick to unsigned gradients for dashcam vehicle detection.

C. Linear SVM Parameters

The optimal Nu value was determined to be 0.09 with the trainAuto function using 10-fold cross-validation. However, although this value of the parameter minimized the testing error, we observed bad detection on real images and videos. After experimenting, we found out that for $Nu \approx 0.25$ the pipeline performs very good and the detections are very accurate. In this case the testing error is larger, but it improves the performance by significantly reducing the false positives on dashcam data. We assume that the vehicles and non-vehicles images from the dataset have more distinguishable HOG feature vectors, so smaller Nu value produces smaller error. On the other hand, sliding windows of a real image that contain or do not contain vehicles seem to have similar HOG feature vectors which requires larger misclassification cost or larger Nu value, thus reducing the number of false positives. So, the Nu value depends on the training data and the input images or videos.

D. detectMultiScale parameters

The optimal values of the detectMultiScale parameters depend on the input on which the detection is performed. The hitThreshold parameter is a threshold for the distance between the features and the SVM classifying plane. Setting its value too low will cause false positives and setting its value too high will lead to false negatives. In our experience, for best results the hitThreshold value should be between 1 and 2 depending on the finalThreshold value and the input image or video, when $Nu = 0.25$. The finalThreshold parameter is an NMS threshold. The function will classify a region as a vehicle if there are more positives in the region than this given threshold, otherwise it will classify the region as non-vehicle. Setting its value too low will cause false positives and setting its value too high will cause to false negatives. In our experience, for best results the finalThreshold value should be between 0.5 and 1.5 depending on the hitThreshold value and the input image or video, for $Nu = 0.25$. Figure 1 shows the results with hitThreshold set to 1.25 and finalThreshold set to 0.75, values that were found to give the desired results.

E. Real Data Performance

Although the described pipeline performs reasonably well on real dashcam images, its performance on real dashcam videos is not satisfying. It is unusual that there are lots of false positives in a video frame, while there are none in an image that seems identical to that video frame. This is the case because video frames contain artifacts which are caused by the application of lossy compression. HOG descriptors describe the shapes or the edges of an object, so the HOG feature vectors of those artifacts can be very similar to the ones of real vehicles. That will cause false positive misclassifications in unusual regions, like in the sky or the asphalt. As mentioned before, gamma correction, larger discrete derivative mask and Gaussian smoothing can be applied on videos to improve the detector's performance. However, vehicle detection in lossily compressed images and videos still suffers from false positives as they are not fully eliminated by tweaking these HOG parameters.

IV. Discussion

The described approach performs reasonably well on images and achieves mediocre performance on videos. There are numerous proposed solutions for this problem. Unfortunately, tuning the HOG parameters is not one of them. Even if there are some optimal values for these parameters, performing detection on every video frame is computationally expensive. Bear in mind that those values will not be optimal for other videos. The right approach would be to implement vehicle

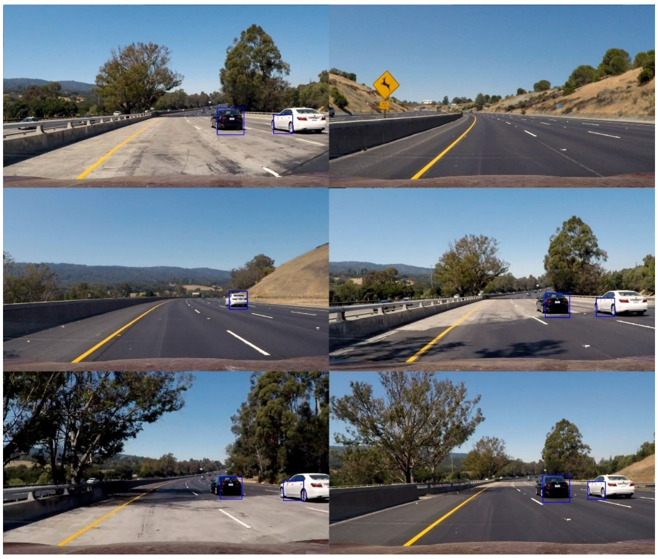


Figure 1. Vehicle detection in images with hitThreshold=1.25 and finalThreshold=0.75 with mean shift grouping

tracking, which is computationally inexpensive, rather than performing detection on every frame.

Some latest trends prefer using CNNs, RNNs or other deep learning approaches for these types of problems. We agree that these approaches are faster and more robust. However, our goal was to use classic image processing techniques instead of neural networks, because we wanted to get some perspective and experience in image processing and computer vision. So, although this approach cannot be used for real-time detection and is susceptible to false positives, it helps in understanding color spaces, image gradients, and SVM classifiers.

V. Conclusion and Future Work

We studied the influence of the color spaces on the performance and concluded that decorrelated and perceptual color spaces work best. We also studied the influence of the HOG parameters on the performance and concluded that in most cases, the optimal values of the HOG parameters for vehicle detection problems are the same as the proposed HOG parameters for human detection problems given in the original HOG paper [1]. HOG, SVM and detectMultiScale parameters are highly correlated and the choice of their values has profound effects on the performance of the detection. However, the performance of these parameters also depends on the input image or video and the goal should be finding the values for these parameters that generally work reasonably well with a lot of real images and videos and allow few misclassifications, instead of finding the perfect values for the parameters for only one image or video. The few misclassifications should be handled with other techniques.

We have shown that using HOG and linear SVM is a viable approach for vehicle detection in images, while it has some limitations for vehicle detection in videos. However, by using some simple techniques and extending the pipeline, this approach can easily overcome these limitations. Implementation of a vehicle tracking system is one of the future steps that will be considered for improving this pipeline. The false positives can be eliminated by checking whether the positive detections in a region are appearing in more consecutive frames. There are numerous HOG extensions and improvements that can be used, and SVMs with more complex kernels or modern and more sophisticated classification algorithms can be considered. These improvements can make the system resistant to artifacts and can provide overall better detection.

References

[1] N. Dalal and B. Triggs, “Histograms of oriented gradients for human detection”, in *international Conference on computer vision & Pattern Recognition (CVPR’05)*, IEEE Computer Society, vol. 1, 2005, pp. 886–893.

[2] I. M. Creusen, R. G. Wijnhoven, E. Herbschleb, and P. de With, “Color exploitation in hog-based traffic sign detection”, in *2010 IEEE International Conference on Image Processing*, IEEE, 2010, pp. 2669–2672.

[3] L. Mao, M. Xie, Y. Huang, and Y. Zhang, “Preceding vehicle detection using histograms of oriented gradients”, in *2010 International Conference on Communications, Circuits and Systems (ICCCAS)*, IEEE, 2010, pp. 354–358.

[4] J. Arróspide, L. Salgado, and M. Camplani, “Image-based on-road vehicle detection using cost-effective histograms of oriented gradients”, *Journal of Visual Communication and Image Representation*, vol. 24, no. 7, pp. 1182–1190, 2013.

[5] R. Girshick, “Fast r-cnn”, in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1440–1448.

[6] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You only look once: Unified, real-time object detection”, in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 779–788.

[7] G. Rossum, “Python reference manual”, Amsterdam, The Netherlands, The Netherlands, Tech. Rep., 1995.

[8] G. Bradski, “The OpenCV Library”, *Dr. Dobb’s Journal of Software Tools*, 2000.

[9] I. P. G. at UPM, *Gti vehicle image database*, https://www.gti.ssr.upm.es/data/, 2011.

[10] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, “Vision meets robotics: The kitti dataset”, *International Journal of Robotics Research (IJRR)*, 2013.

ISBN 9786084699095

9 786084 699095